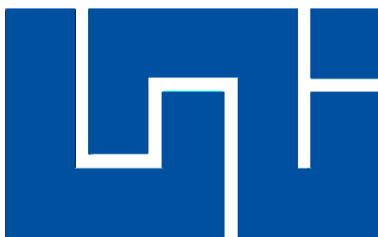


# **Universidad Nacional de Ingeniería**

## **Facultad de Electrotecnia y Computación**



### **MANUAL DE USUARIO Y TÉCNICO**

#### **SISTEMA WEB DE SOLICITUD DE TRÁMITES ACADÉMICOS PARA LA FACULTAD DE ELECTROTECNIA Y COMPUTACIÓN F.E.C**

**AUTOR:**

Br.Marcos Antonio Boza Saldívar 2008-23166

**TUTOR:**

Msc. Gloria Thalía Flores Quintana

Managua, 28 de Septiembre del 2015

## CONTENIDO

1. INTRODUCCIÓN.....	5
2. TIPOS DE CUENTA.....	6
3. PANTALLA DE ACCESO.....	6
3.1 Registro.....	7
3.2 Recuperación de contraseñas.....	8
3.3 Página de Inicio .....	8
4. OPCIÓN DE CONTÁCTENOS.....	10
5. CAMBIO DE CONTRASEÑA.....	11
6. CREACIÓN DE REQUISITOS .....	12
6.1 Edición de Requisitos .....	13
7. CREACIÓN DE TRÁMITES.....	13
7.1 Edición de trámites.....	15
8. CREACIÓN DE ÁREAS .....	15
8.1 Edición de Áreas.....	16
9. CREACIÓN DE CARRERAS.....	17
9.1 Edición de Carreras.....	18
10. CREACIÓN DE FACULTAD.....	19
10.1 Edición de Facultades.....	20
10.2 Creación de Recinto.....	20
10.3 Edición de Recintos.....	21
11. MOVIMIENTOS .....	22
11.1 Bandeja de Entrada.....	22
11.1 Cuentas de Usuario.....	24
11.2 Activación de Cuentas .....	25
12. HISTORIAL DE TRÁMITES.....	27
13. CATALOGO.....	28
13.1 Edición de Catálogos .....	29
14. ADMINISTRACIÓN .....	30
14.1 Perfiles .....	30
14.2 Edición de perfil .....	31
14.3 Personal Administrativo.....	32

14.4	Edición de personal administrativo .....	32
14.5	Asignación de Perfiles .....	33
14.6	Edición de asignación de perfil.....	34
15.	REPORTES ADMINISTRATIVOS .....	34
15.1	Acceso de Usuarios .....	35
15.2	Cambio de clave .....	36
15.3	Historial de Tramites .....	37
15.4	Reporte de Transacciones .....	38
15.5	Reporte grafico de porcentaje de trámites .....	40
16.	CONFIGURACIÓN ADMINISTRATIVA .....	41
16.1	Agregar nuevas Opciones.....	41
16.2	Edición de Nueva Opción.....	42
16.3	Configuración de Opciones por Perfil.....	43
16.4	Configuración de Sesión .....	44
17.	MOVIMIENTOS .....	44
17.1	Edición de Requisitos .....	45
18.	TRAMITE DIRECTO .....	46
19.	NUEVO TRAMITE .....	50
20.	INTRODUCCIÓN.....	55
21.	DISEÑO DE INTERFAZ .....	56
22.	OPCIONES DE MENU .....	57
23.	ESTRUCTURA DE DATOS .....	59
24.	CLASES PRINCIPALES DE LA APLICACIÓN.....	60
24.1	CONEXIÓN EN EL SISTEMA WEB.....	60
24.2	CLASES DE ACCESO .....	61
25.	CRUD PARA EL SISTEMA WEB .....	69
25.1	CRUD DE TRÁMITES ACADÉMICOS .....	69
25.2	DISEÑO DEL FORMULARIO DE WFP_documento.aspx.....	69
25.3	INSERCIÓN.....	74
25.3.1	FUNCIÓN DE INSERCIÓN DE NUEVOS DATOS. ....	75
25.3.2	FUNCIÓN DE INSERCIÓN LINQ .....	76
25.4	EDICIÓN .....	76

25.4.1	FUNCION DE EDICION DE DATOS.....	77
25.4.2	FUNCION DE INSERCION LINQ .....	78
25.4.3	VISUALIZACION DE DATOS .....	78
25.4.4	FUNCION DE VISUALIZACION.....	78
26.	FUNCION DE VISUALIZACION LINQ .....	79
27.	PROCESAMIENTO DE TRAMITES ACADEMICOS .....	79
27.1	FUNCION DE VERIFICACION DE PERFIL .....	79
27.2	FUNCION PARA PROCESAR TRÁMITES .....	80
28.	REALIZAR UN NUEVO TRAMITE.....	82
28.1	WIZARD .....	82
28.2	CONSTRUCCION DEL WIZARD .....	83
28.3	CONCLUSIÓN DE TRÁMITE .....	93
28.4	CONFIRMACION DEL TRÁMITE .....	95
28.5	INSERCION DEL TRÁMITE .....	96
29.	REPORTES.....	97
29.1	REPORTE DE HISTORIAL DE TRÁMITES.....	97



## 1. INTRODUCCIÓN

El sistema web de trámites académicos para facultad de electrotecnia y computación F.E.C fue creado con el fin de que los estudiantes puedan solicitar y cancelar los trámites académicos vía online de manera rápida y segura y para que el personal administrativo pueda realizar una revisión periódica brindando un seguimiento durante las etapas de los trámites y tener un mejor control, orden y pagos de los trámites académicos que realizan los estudiantes desde las diferentes métodos (online, presencial y bancaria) que convergen en una misma bandeja.

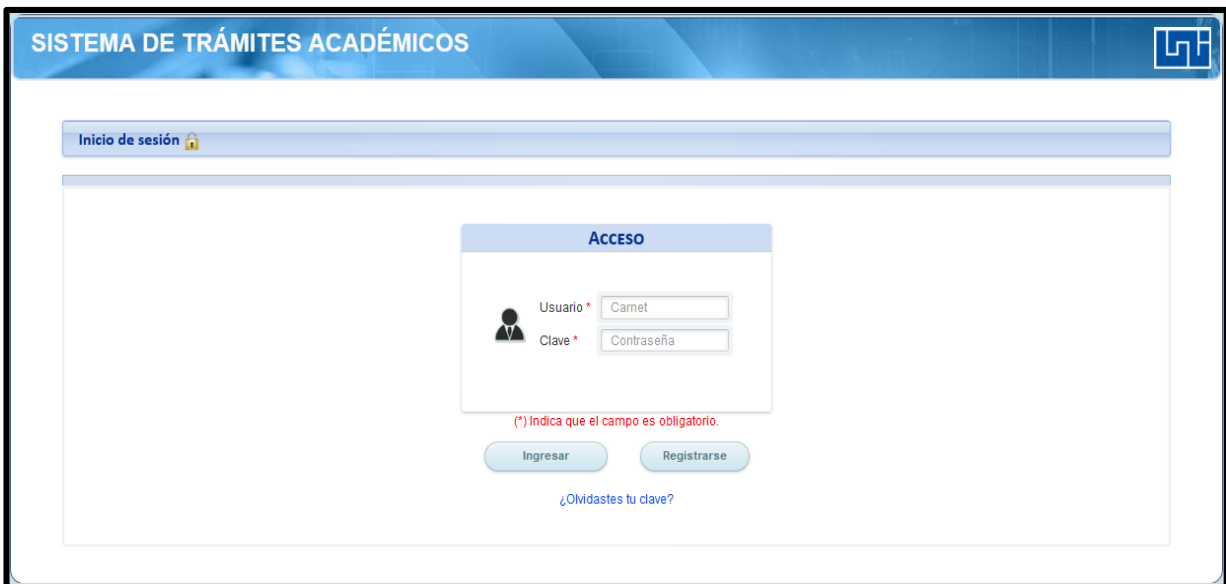
Los estudiantes de la facultad de electrotecnia y computación (FEC) podrán tener acceso a sus cuentas previamente registradas, y luego activadas por el administrador, podrán solicitar trámites académicos de forma online, llevar un seguimiento de las solicitudes (vía online, de forma presencial o por procesos bancarios) todas en una bandeja con sus respectivos detalles, realizar los pagos por tarjetas de crédito o débito y tener un historial de todos los trámites realizados.

## 2. TIPOS DE CUENTA

En el sistema se manejan 3 tipos de cuentas de usuarios, donde cada una cuenta con un nivel de privilegios y restricciones diferentes, esto se hace por políticas de seguridad, integridad y privacidad de la información.

## 3. PANTALLA DE ACCESO

La pantalla de inicio del sistema web de trámites académicos donde se podrán visualizar las opciones de **registro**, **recuperación de contraseña** y **acceso** a las cuentas.



The screenshot shows the login interface of the 'SISTEMA DE TRÁMITES ACADÉMICOS'. At the top, there is a blue header with the system name and a logo. Below the header, a light blue bar contains the text 'Inicio de sesión' with a lock icon. The main content area features a central white box titled 'ACCESO'. Inside this box, there are two input fields: 'Usuario \*' with the placeholder 'Carnet' and 'Clave \*' with the placeholder 'Contraseña'. Below these fields, a red note states '(\*) Indica que el campo es obligatorio.' At the bottom of the box are two buttons: 'Ingresar' and 'Registrarse'. Below the box, there is a blue link that says '¿Olvidastes tu clave?'.

- Esta pantalla es la inicial para los 3 tipos de cuentas.
- Se tiene que ingresar para el caso de los estudiantes: **El número de carnet** y **su contraseña** y luego el botón **“Ingresar”**
- En caso de que requiera registrarse en el sistema se tiene que dirigirse hacia el botón **“Registrarse”**
- Si se necesita recuperar la contraseña se tiene que dirigir al link que dice **“¿Olvidaste tu clave?”**

### 3.1 Registro

Para registrarse en el sistema, para el caso de los estudiantes tienen que estar previamente creados y activos en los registros de la Universidad Nacional de Ingeniería. Para el caso del personal administrativo tiene crearlo el **administrador** directamente en el sistema y activar su cuenta.

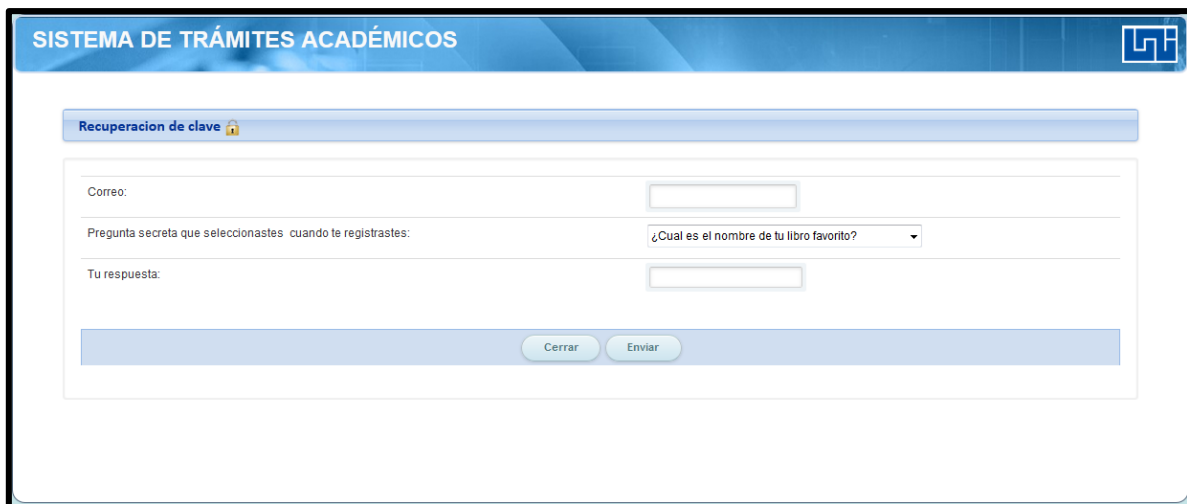
The screenshot shows a web interface for the 'SISTEMA DE TRÁMITES ACADÉMICOS'. The main heading is 'Registro' with a lock icon. Below this, there is a form with the following fields: 'Número de Carnet:' with a text input; 'Clave:' with a text input; 'Correo:' with a text input; 'Elige una pregunta secreta:' with a dropdown menu labeled 'Seleccione una Opción'; and 'Tu respuesta:' with a text input. At the bottom of the form are two buttons: 'Regresar' and 'Guardar'.

- Para empezar el registro en el caso de los estudiantes, se necesita ingresar el **Numero de carnet** (La cuenta de estudiante en la Universidad Nacional de Ingeniería tiene que estar activa).
- Una **clave** personalizada que tiene que cumplir cierto criterio de seguridad (más de 6 caracteres, al menos 1 número y 1 carácter especial).
- Un **correo**, este correo tiene que ser personal debido a que las notificaciones de la conclusión de los trámites se enviaran al correo registrado en esta etapa.
- La **pregunta de seguridad y respuesta** es un método con doble funcionalidad y es integrado en la mayoría de los procesos de registro para aumentar la seguridad en el registro y facilitar la recuperación de contraseñas debido a que la pregunta y su respuesta serán ocupadas como un indicio en el proceso de recuperación de las cuentas.

- Si la operación se realizó exitosamente se mostrara una notificación indicando el éxito de la operación de lo contrario mostrara una alerta.

### 3.2 Recuperación de contraseñas

La recuperación de cuentas consiste en un proceso de seguridad donde el usuario utiliza un método alternativo para que el sistema envíe su contraseña al correo ingresado en el proceso de registro, dicho proceso se llevara a cabo si y solo si la pregunta y la respuesta coincidan.



- El **correo** tiene que ser una personal que se revise periódicamente.
- Las **preguntas secretas** ya están definidas, se tiene que elegir la misma pregunta que se eligió en el proceso de registro.
- La **respuesta** tiene que coincidir con la respuesta que se almacenó en el proceso de registro.
- Cuando la operación se realice exitosamente se mostrara una notificación que indica la revisión del correo que contendrá la contraseña actual para el acceso.

### 3.3 Página de Inicio

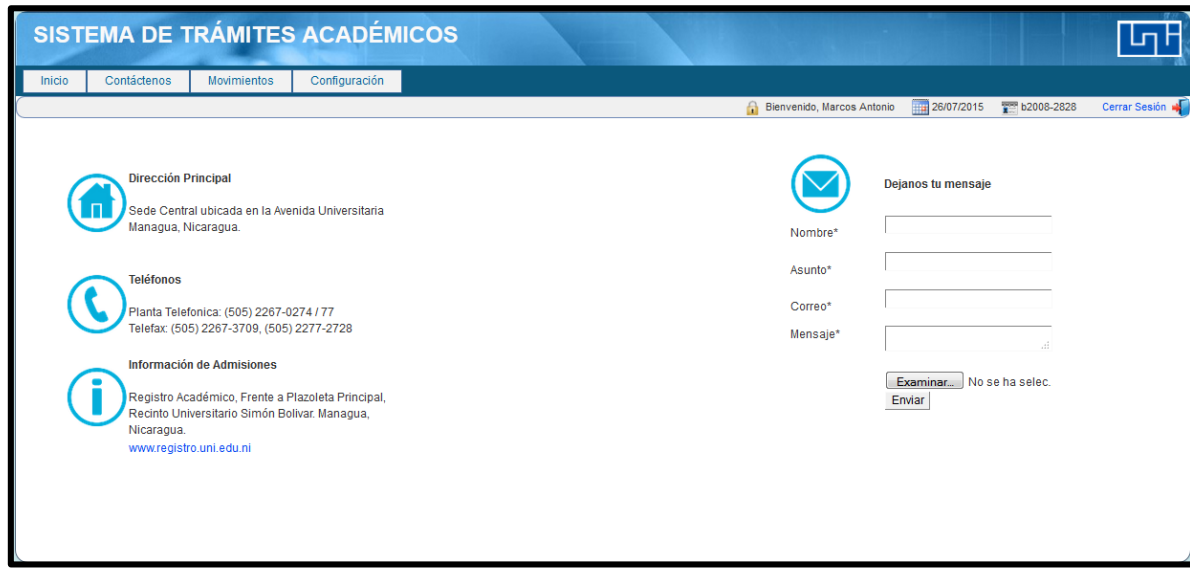
En la página de inicio del sistema se muestra un resumen de las características del sistema web, tipos de pagos, ventajas, uso y pasos, al igual se muestra el menú correspondiente tipo de usuario en dependencia al tipo de Usuario.



- Se muestran en la barra de notificaciones datos personales el usuario como son el número de carnet, fecha actual, Nombre del usuario y con un icono de un mensaje no abierto la cantidad de solicitudes que están recibidas o no han sido enviadas a los encargados de carrera en la secretaria de la facultad.

## 4. OPCIÓN DE CONTÁCTENOS

La opción de “**contáctanos**”, aquí se muestran los datos más importantes para comunicarse con el personal de soporte ante cualquier altercado o solicitud de información.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' website. The top navigation bar includes 'Inicio', 'Contáctenos', 'Movimientos', and 'Configuración'. The user is logged in as 'Marcos Antonio' on '26/07/2015' with ID 'b2008-2628'. The 'Contáctenos' section is active, displaying contact information and a message form.

**Dirección Principal**  
Sede Central ubicada en la Avenida Universitaria  
Managua, Nicaragua.

**Teléfonos**  
Planta Telefonica: (505) 2267-0274 / 77  
Telefax: (505) 2267-3709, (505) 2277-2728

**Información de Admisiones**  
Registro Académico, Frente a Plazoleta Principal,  
Recinto Universitario Simón Bolívar, Managua,  
Nicaragua.  
[www.registro.uni.edu.ni](http://www.registro.uni.edu.ni)

**Dejanos tu mensaje**

Nombre\*

Asunto\*

Correo\*

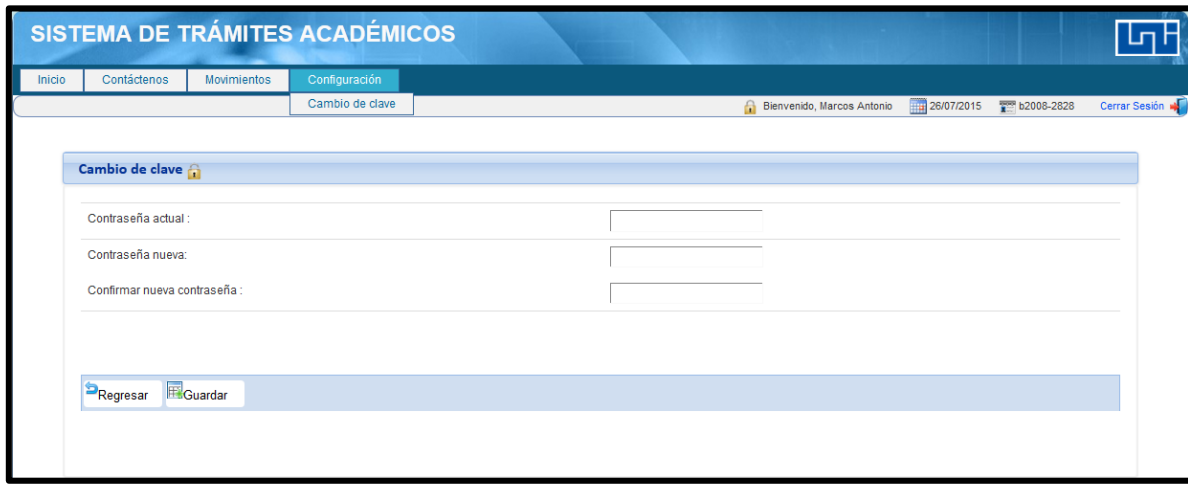
Mensaje\*

No se ha selec.

- Dirección, Teléfonos y Re direccionamiento a la página principal de registro académicos de la facultad.
- **Envío de Correo:** El envío de correo es para que los usuarios puedan enviar sugerencias, reportar algún error teniendo la opción de adjuntar imágenes para una mayor comprensión.

## 5. CAMBIO DE CONTRASEÑA

El cambio de contraseña es un requerimiento en cualquier cuenta de usuario, cabe destacar que este cambio se hace en tiempo real recomendando reiniciar sesión para observar el cambio.

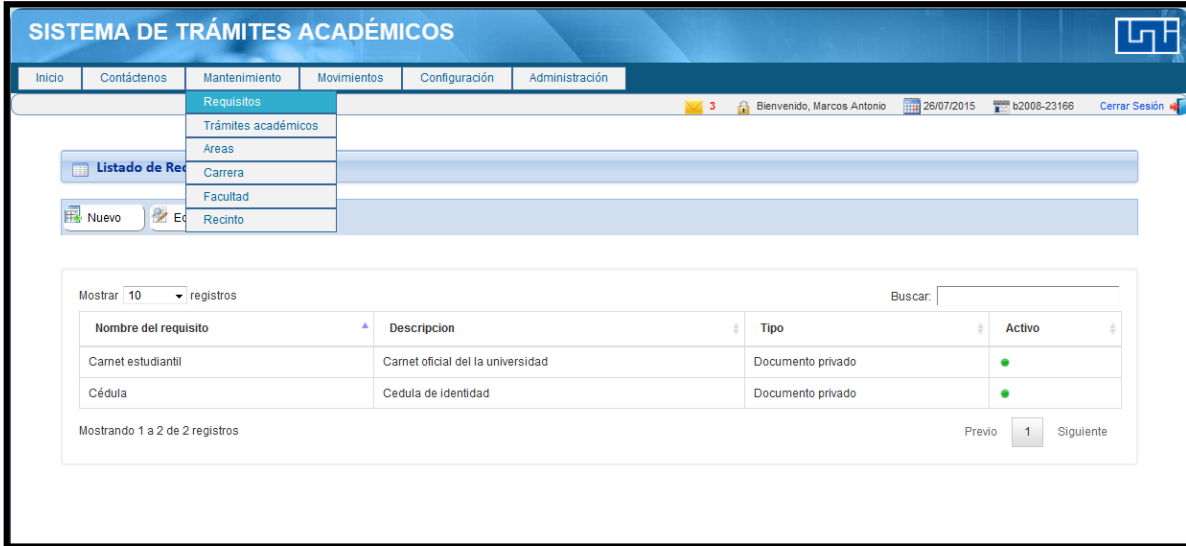


The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The top navigation bar includes 'Inicio', 'Contáctenos', 'Movimientos', and 'Configuración'. The 'Configuración' tab is active, and the 'Cambio de clave' sub-tab is selected. The main content area is titled 'Cambio de clave' and contains three input fields: 'Contraseña actual:', 'Contraseña nueva:', and 'Confirmar nueva contraseña:'. Below these fields are two buttons: 'Regresar' and 'Guardar'. The top right of the interface shows a user greeting 'Bienvenido, Marcos Antonio', the date '26/07/2015', a session ID 'b2008-2828', and a 'Cerrar Sesión' link.

- Se requiere ingresar la **contraseña actual** con la que se accede al sistema.
- Ingresar la **nueva contraseña**, que será sustituirá a la antigua contraseña.
- **Confirmación de nueva contraseña**, se tiene que ingresar la misma contraseña nueva.
- Si las contraseñas no coinciden aparecerá una notificación que indica un error en las contraseñas, se recomienda tener mucho cuidado y poner contraseñas no muy complicadas para evitar olvidarlas.

## 6. CREACIÓN DE REQUISITOS

Los requisitos son documentos necesarios para realizar la solicitud de los trámites académicos de manera presencial, para la creación de un requisito se necesita realizar los siguientes pasos:

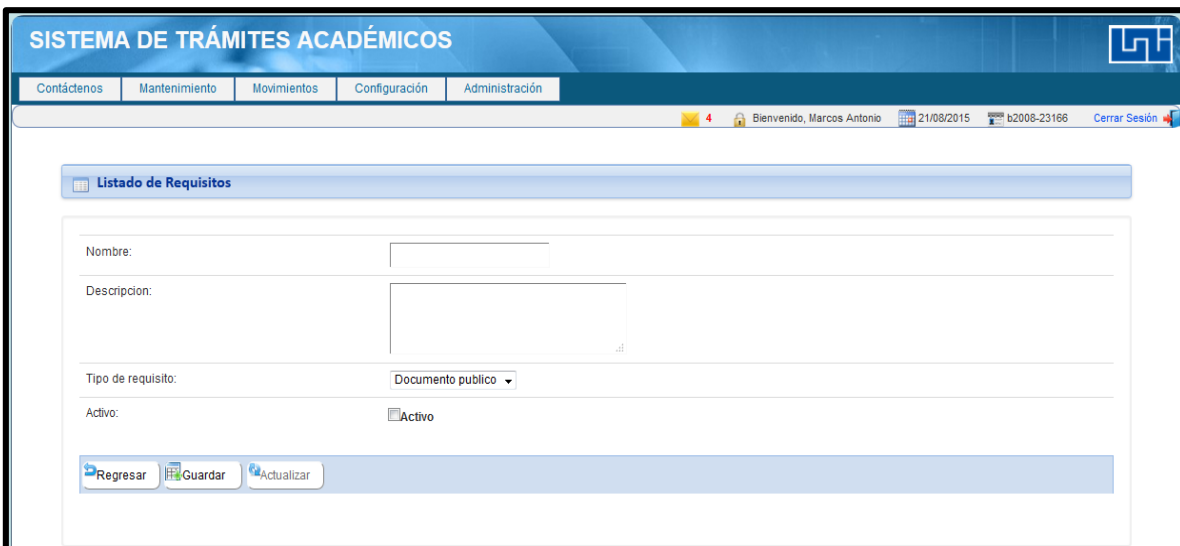


The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The 'Requisitos' menu is expanded, showing options like 'Trámites académicos', 'Áreas', 'Carrera', 'Facultad', and 'Recinto'. The 'Nuevo' button is visible. Below the menu, a table lists existing requirements:

Nombre del requisito	Descripción	Tipo	Activo
Carnet estudiantil	Carnet oficial de la universidad	Documento privado	<input checked="" type="checkbox"/>
Cédula	Cedula de identidad	Documento privado	<input checked="" type="checkbox"/>

Mostrando 1 a 2 de 2 registros. Navigation: Previa 1 Siguiente.

- Dirigirse a la opción “**Mantenimiento**”, se desplegará un lista de opciones, elegir la opción “**Requisitos**”.
- Se desplegara un formulario donde se presentan el nombre, descripción, tipo y estado del registro.
- Para crear uno nuevo presionar el botón **Nuevo**



The screenshot shows the 'Listado de Requisitos' form. It includes fields for 'Nombre:', 'Descripción:', 'Tipo de requisito:' (with a dropdown menu set to 'Documento publico'), and 'Activo:' (with a checkbox). At the bottom, there are buttons for 'Regresar', 'Guardar', and 'Actualizar'.



- Se deben ingresar los datos requeridos, de lo contrario aparecerá un mensaje notificando que hay campos vacíos.
- Después de llenar los campos correctamente, presionar el botón **Guardar** que devolverá al formulario anterior mostrando el registro que se ingresó y una notificación indicando que el registro fue almacenado.

## 6.1 Edición de Requisitos

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



Nombre del requisito	Descripción	Tipo	Activo
Carnet estudiantil	Carnet oficial de la universidad	Documento privado	●
Cédula	Cedula de identidad	Documento privado	●

- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

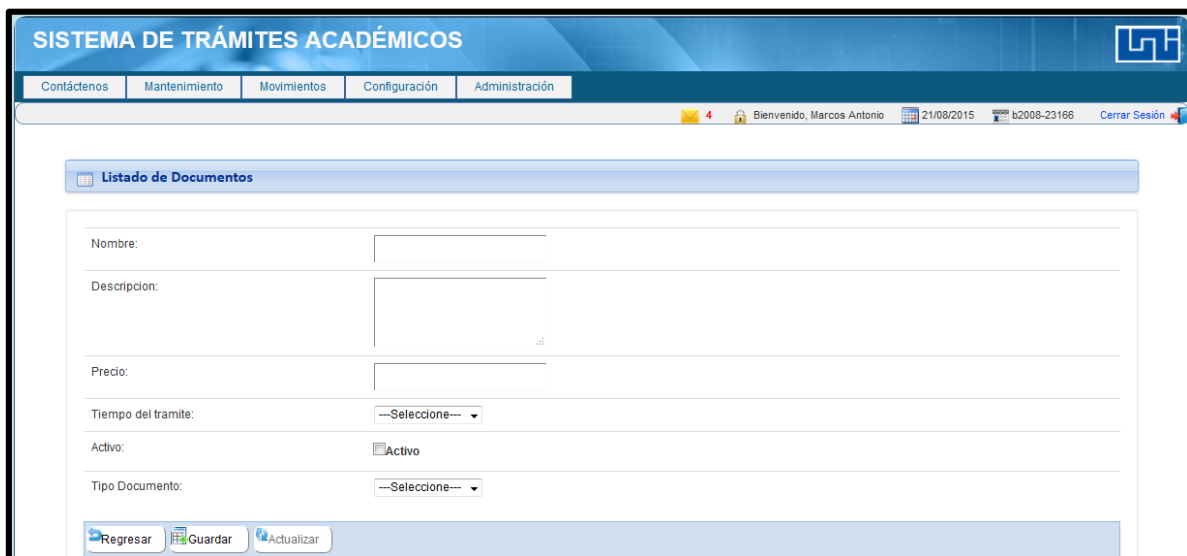
## 7. CREACIÓN DE TRÁMITES

La creación de trámites es uno de los formularios más importantes del sistema web, debido a que aquí se crean los documentos que podrán ser solicitados por los estudiantes, se determinará el precio y duración que tarda dicho proceso.



Nombre del documento	Tipo de documento	Precio	Tiempo	Activo
Carnet	Carnet	110	15 dias	●
Carta de egresado	Carta	75	1 dia	●

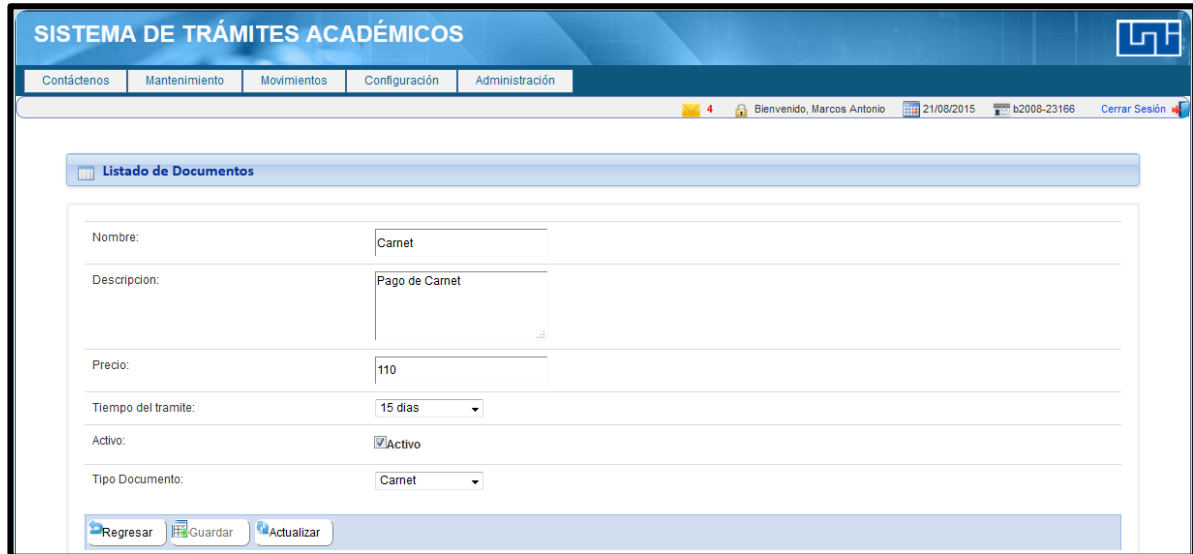
- Dirigirse a la opción “**Mantenimiento**”, se desplegará una lista de opciones luego elegir “**Trámites académicos**”.
- Se desplegara un formulario donde se presentan el nombre del documento, Tipo del documento, precio, el tiempo de trámite y estado del registro.
- Para crear un nuevo documento, presionar el botón **Nuevo**.



- Luego de rellenar los campos requeridos, presionar el botón **Guardar** para crear el nuevo registro.
- Aparecerá una notificación para indicar que el registro se creó con éxito, de lo contrario aparecerá una notificación que indica un problema con la creación del registro.

## 7.1 Edición de trámites

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 21/08/2015 | b2008-23166 | Cerrar Sesión

**Listado de Documentos**

Nombre:

Descripción:

Precio:

Tiempo del trámite:

Activo: ☒ Activo

Tipo Documento:

- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 8. CREACIÓN DE ÁREAS

Este formulario permite el anexo de áreas administrativas para indicar a que área serán dirigidas las solicitudes de los trámites, esta parte es algo referencial.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Inicio | Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 26/07/2015 | b2008-23166 | Cerrar Sesión

**Listado de Áreas**

Nombre del area	Descripción	Activo
Diñ	Diñ	●
Registro	registro academico	●
Tesoreria	Caja	●

Mostrando 1 a 3 de 3 registros

Previo  Siguiente

- Primeramente dirigirse a la opción del menú “**Mantenimiento**”.
- Luego elegir la opción “**Áreas**”.
- Para crear una nueva área, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 21/08/2015 | b2008-23166 | Cerrar Sesión

**Listado de Areas**

Nombre:

Descripción:

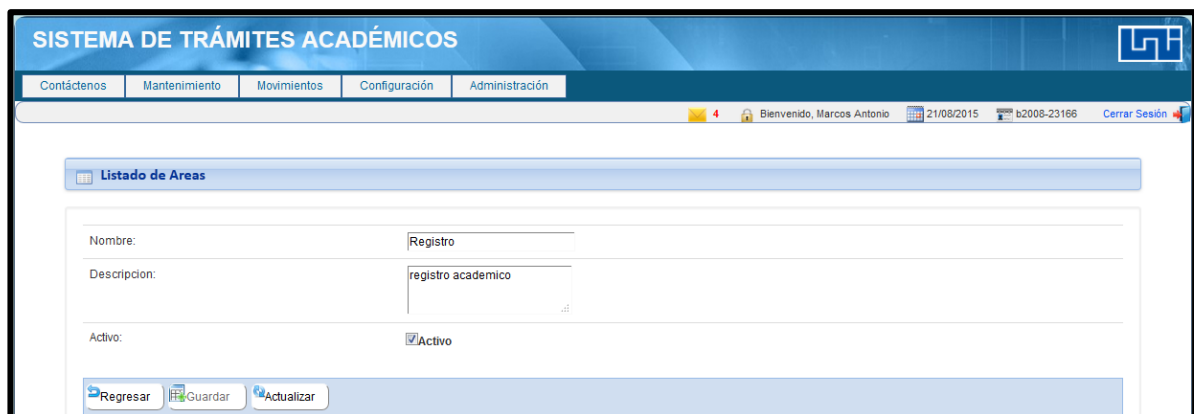
Activo: ☒ Activo

Regresar | Guardar | Actualizar

- Después de guardar, se desplegara el formulario anterior con el registro que se almaceno correctamente.

## 8.1 Edición de Áreas

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 21/08/2015 | b2008-23166 | Cerrar Sesión

**Listado de Areas**

Nombre:

Descripción:

Activo: ☒ Activo

Regresar | Guardar | Actualizar

- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 9. CREACIÓN DE CARRERAS

Las carreras son otro factor importante en el sistema, debido a que tienen que estar registradas todas carreras a las que se les dará cobertura en el sistema.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The 'Mantenimiento' menu is active, and the 'Carrera' option is selected. The main content area displays a table of existing careers.

Nombre de la carrera	Descripción	Activo
Ing. Computacion	Ingeniería en computacion	●
Ing. Electrica	Ingeniería Electrica	●
Ing. en Sistemas	Ingeniería en sistemas	●

Mostrando 1 a 3 de 3 registros

Previo 1 Siguiente

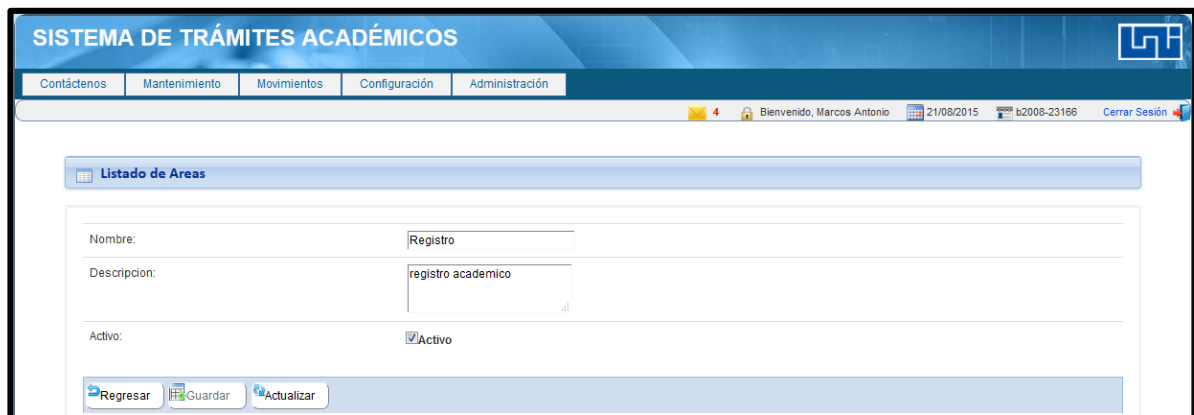
- Primeramente dirigirse a la opción del menú “**Mantenimiento**”.
- Luego elegir la opción “**Carreras**”.
- Para crear una nueva área, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.



- Después de guardar, se desplegará el formulario anterior con el registro que se almacene correctamente.

## 9.1 Edición de Carreras

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



- Se cargará el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 10. CREACIÓN DE FACULTAD

La creación de facultades es importante en el sistema ya que estas definen el límite que abarcara el sistema web en cuanto a facultades de la universidad.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The 'Mantenimiento' menu is open, and 'Facultad' is selected. The main area displays a table of existing faculties:

Nombre de la facultad	Descripción	Activo
FAQ	Facultad de Química	<input checked="" type="checkbox"/>
FEC	Facultad de electrónica y computación	<input checked="" type="checkbox"/>

Below the table, it says 'Mostrando 1 a 2 de 2 registros'. Navigation buttons 'Previo' and 'Siguiente' are visible.

- Primeramente dirigirse a la opción del menú “**Mantenimiento**”.
- Luego elegir la opción “**Facultad**”.
- Para crear una nueva área, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.



The screenshot shows the 'Listado de Facultades' form. It contains the following fields:

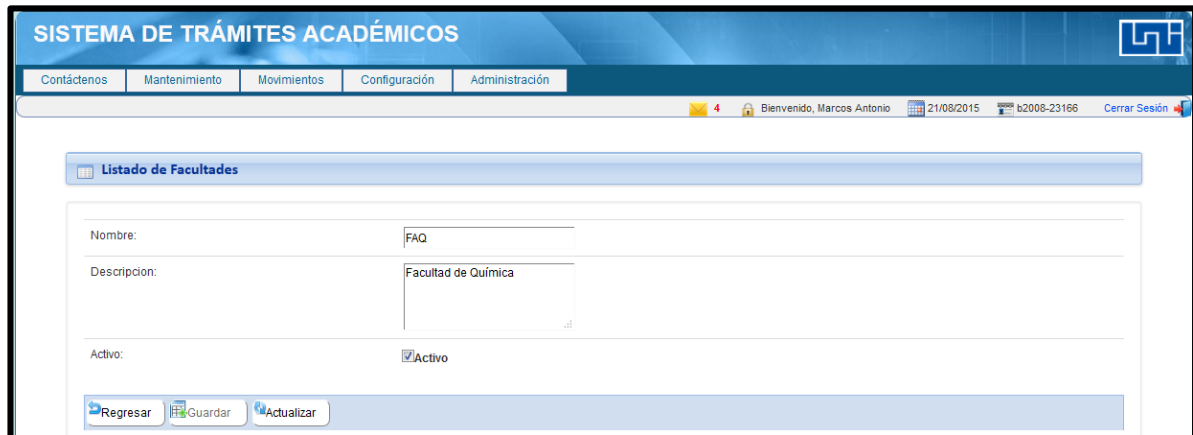
- Nombre:
- Descripción:
- Activo: ☒ Activo

At the bottom, there are three buttons: 'Regresar', 'Guardar', and 'Actualizar'.

- Después de guardar, se desplegara el formulario anterior con el registro que se almaceno correctamente.

## 10.1 Edición de Facultades

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



The screenshot shows the 'Listado de Facultades' form. It has a header with navigation tabs: 'Contáctenos', 'Mantenimiento', 'Movimientos', 'Configuración', and 'Administración'. Below the header, there's a user greeting 'Bienvenido, Marcos Antonio' and a date '21/08/2015'. The form itself has three input fields: 'Nombre:' with the value 'FAQ', 'Descripción:' with the value 'Facultad de Química', and 'Activo:' with a checked checkbox. At the bottom, there are three buttons: 'Regresar', 'Guardar', and 'Actualizar'.

- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 10.2 Creación de Recinto

La creación de facultades es importante en el sistema ya que estas definen el límite que abarcara el sistema web en cuanto a facultades den la universidad.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The 'Mantenimiento' menu is open, showing options: 'Requisitos', 'Trámites académicos', 'Áreas', 'Carrera', 'Facultad', and 'Recinto'. The 'Recinto' option is selected. Below the menu, there's a table with the following data:

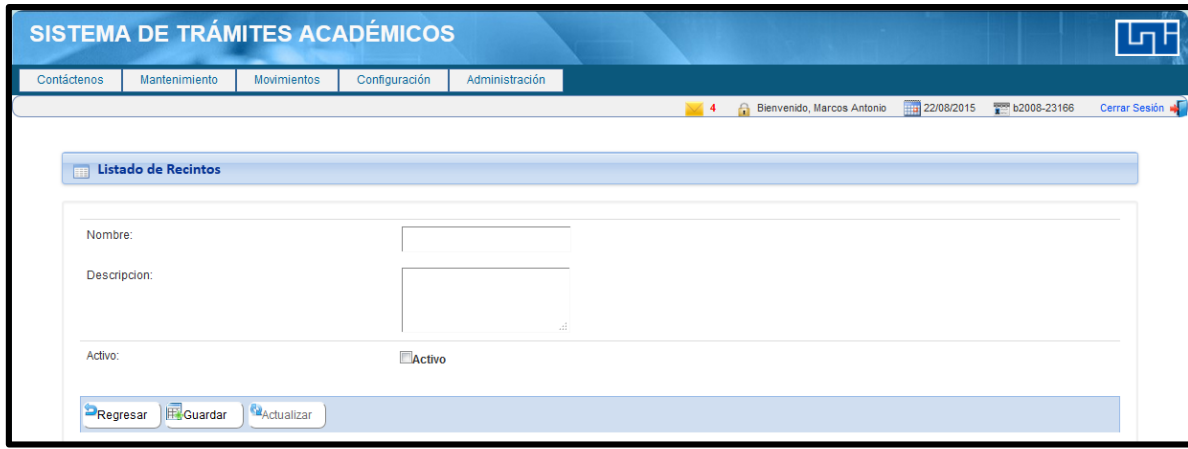
Nombre de la Recinto	Descripción	Activo
RUPAP	RUPAP	<input checked="" type="checkbox"/>
RUSB	Recinto universitario Simon Bolivar	<input checked="" type="checkbox"/>

At the bottom of the table, it says 'Mostrando 1 a 2 de 2 registros'. There are also navigation buttons: 'Previo', '1', and 'Siguiete'.

- Primeramente dirigirse a la opción del menú “**Mantenimiento**”.



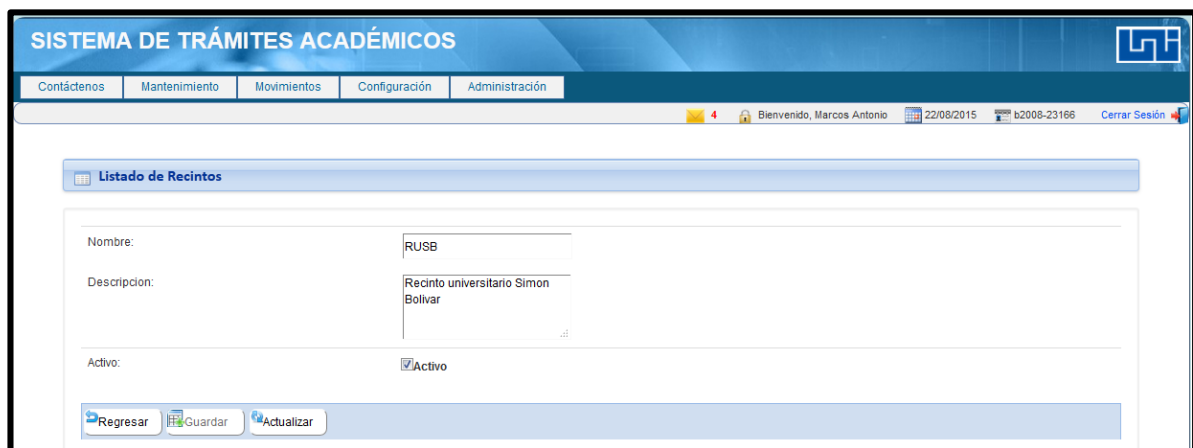
- Luego elegir la opción “**Recinto**”.
- Para crear una nueva área, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.



- Después de guardar, se desplegara el formulario anterior con el registro que se almaceno correctamente.

### 10.3 Edición de Recintos

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 11.MOVIMIENTOS

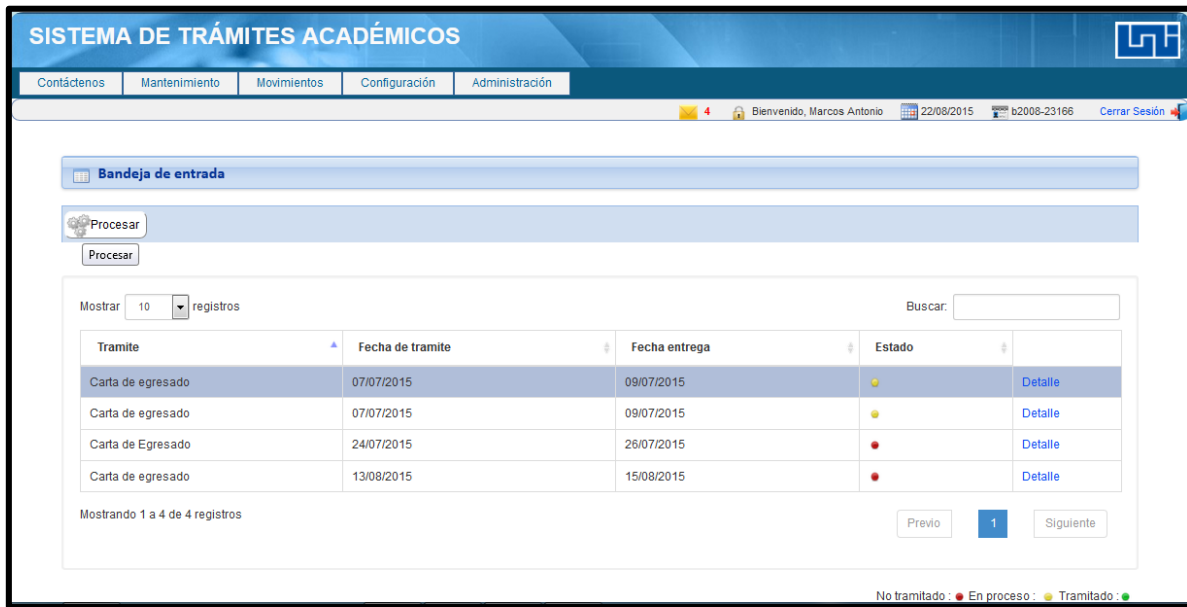
En esta sección se encuentran todas las opciones referentes a las solicitudes Y al manejo de cuentas de los usuarios. Se desplegaran las opciones de Bandeja de entrada, Cuentas e Historial de trámites.



### 11.1 Bandeja de Entrada

Esta opción re direccionará al formulario que muestra la lista de trámites que solicitados y que fueron enviados a la bandeja del encargado de cada carrera. Para procesar las solicitudes, se debe seleccionar la fila que contiene el registro que se requiere cambiar de estado:

- Seleccione la fila del registro a procesar, haciendo **click izq.** sobre la fila.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Bandeja de entrada**

Procesar

Mostrar 10 registros

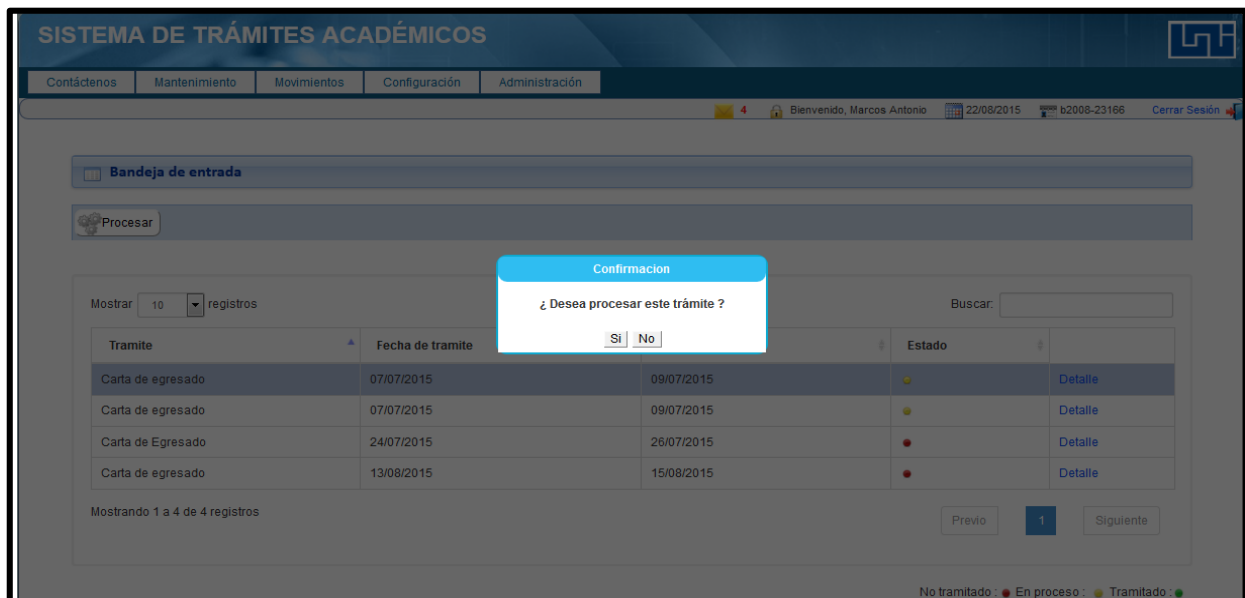
Tramite	Fecha de tramite	Fecha entrega	Estado	
Carta de egresado	07/07/2015	09/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	07/07/2015	09/07/2015	●	<a href="#">Detalle</a>
Carta de Egresado	24/07/2015	26/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	13/08/2015	15/08/2015	●	<a href="#">Detalle</a>

Mostrando 1 a 4 de 4 registros

Previo 1 Siguiente

No tramitado : ● En proceso : ● Tramitado : ●

- En este formulario se encuentra un botón **“Procesar”**, que procesara el registro que se haya seleccionado en el paso anterior.
- Cuando presione el botón **“Procesar”**, se mostrara un mensaje para confirmar si usted procesará el trámite.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Bandeja de entrada**

Procesar

Mostrar 10 registros

¿Desea procesar este trámite?

Si No

Tramite	Fecha de tramite	Fecha entrega	Estado	
Carta de egresado	07/07/2015	09/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	07/07/2015	09/07/2015	●	<a href="#">Detalle</a>
Carta de Egresado	24/07/2015	26/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	13/08/2015	15/08/2015	●	<a href="#">Detalle</a>

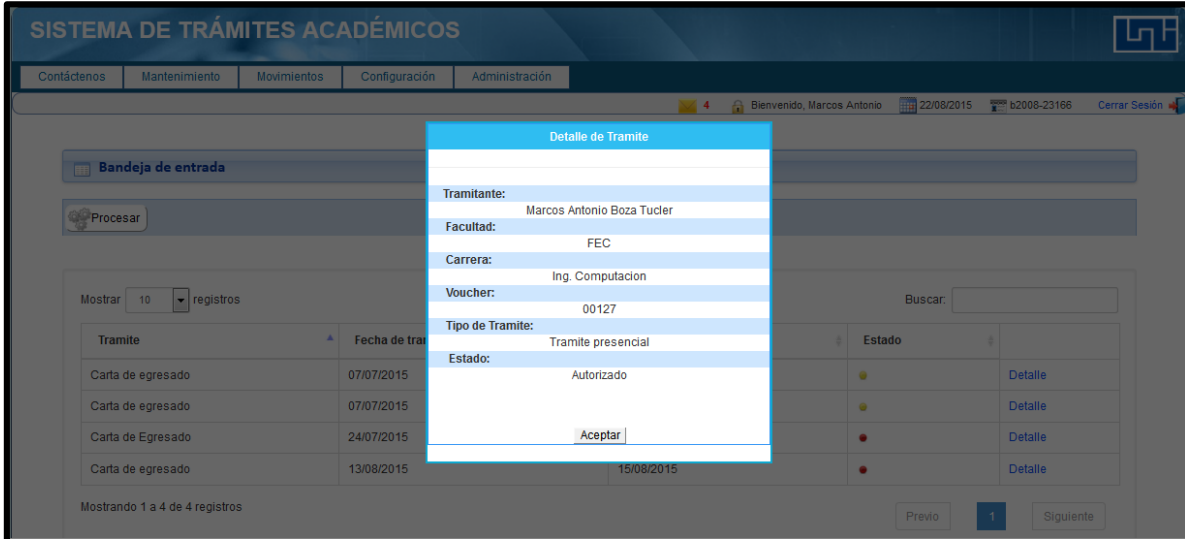
Mostrando 1 a 4 de 4 registros

Previo 1 Siguiente

No tramitado : ● En proceso : ● Tramitado : ●

- Si acepta, el estado del trámite seleccionado cambiará a otro según el estado anterior, en cambio si no acepta el estado del trámite no cambiara.

- En cada fila que contiene un registro, existe una opción “Ver Detalle” en esta opción se mostrara información importante del solicitante y del trámite.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. A modal window titled 'Detalle de Trámite' is open, displaying the following information:

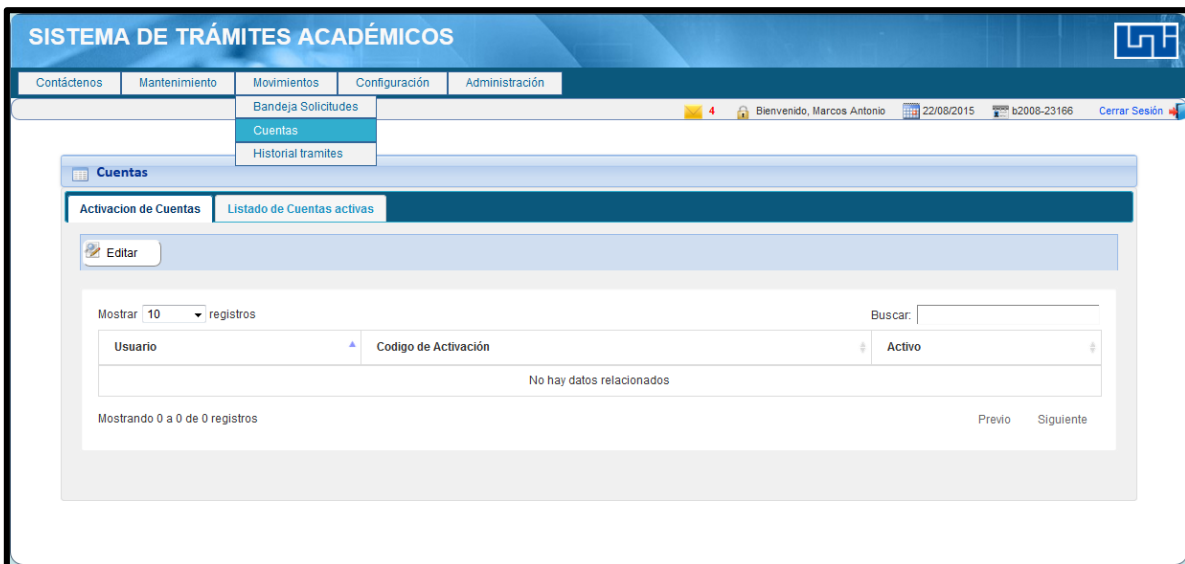
Tramitante:	Marcos Antonio Boza Tucler
Facultad:	FEC
Carrera:	Ing. Computacion
Voucher:	00127
Tipo de Trámite:	Trámite presencial
Estado:	Autorizado

Below the modal, a table of records is visible:

Trámite	Fecha de trámite	Estado	Detalle
Carta de egresado	07/07/2015	Autorizado	<a href="#">Detalle</a>
Carta de egresado	07/07/2015	Autorizado	<a href="#">Detalle</a>
Carta de Egresado	24/07/2015	Autorizado	<a href="#">Detalle</a>
Carta de egresado	13/08/2015	Autorizado	<a href="#">Detalle</a>

## 11.1 Cuentas de Usuario

Esta opción re direccionará al formulario que muestran 2 formularios divididos, uno para las cuentas que no están activas (estudiantes recién registrados o que se les cancelo la cuenta) y que se pueden activar y la otra es un historial de las cuentas activas donde se pueden desactivar.

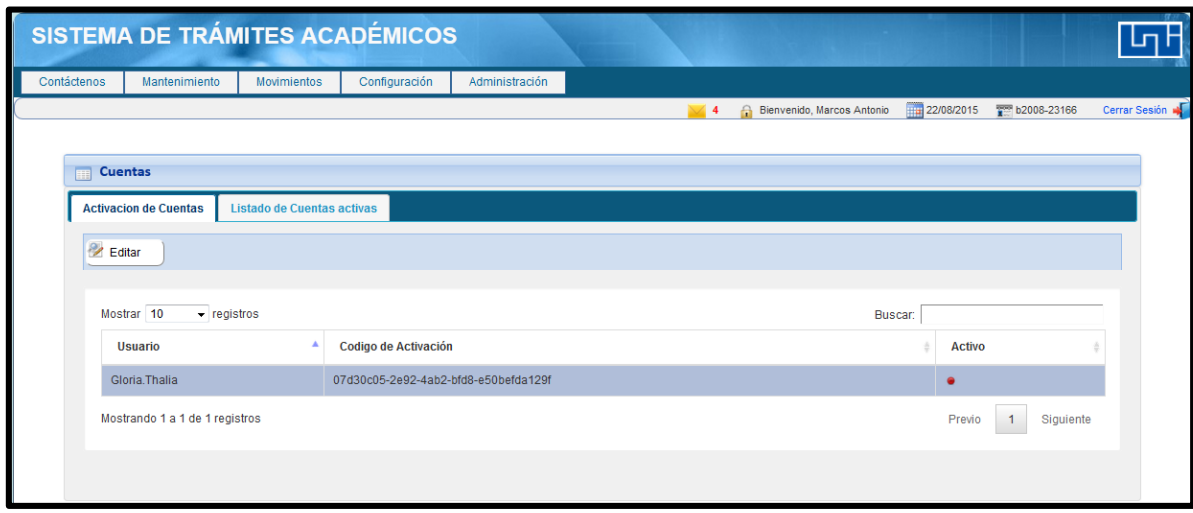


The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface with the 'Cuentas' section selected. The 'Listado de Cuentas activas' tab is active, displaying a table with the following columns:

Usuario	Codigo de Activación	Activo
No hay datos relacionados		

The interface also includes a search bar and pagination controls at the bottom.

## 11.2 Activación de Cuentas



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Cuentas**

Activación de Cuentas | Listado de Cuentas activas

Editar

Mostrar 10 registros

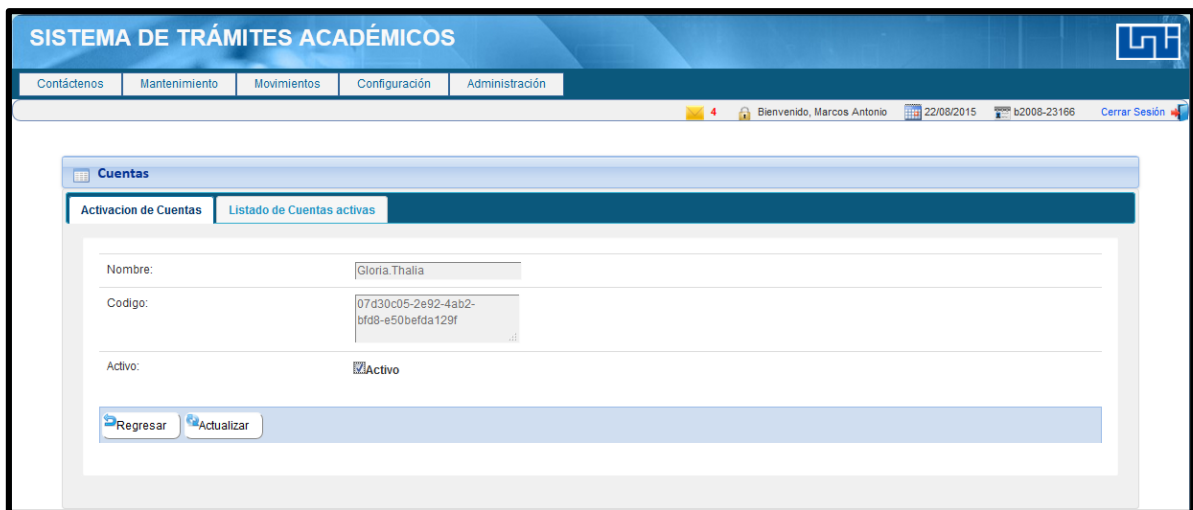
Buscar:

Usuario	Codigo de Activación	Activo
Gloria.Thalia	07d30c05-2e92-4ab2-bfd8-e50befda129f	<input checked="" type="checkbox"/>

Mostrando 1 a 1 de 1 registros

Previo 1 Siguiente

- Para activar una cuenta se tiene que **seleccionar** la fila que contiene el registro de la cuenta que se requiere activar.
- Después de seleccionar el registro, presionar el botón **“Editar”**.
- Se cargara contenido del registro y el campo **activo** desactivado.
- Cambiar el estado del campo **activo**.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Cuentas**

Activación de Cuentas | Listado de Cuentas activas

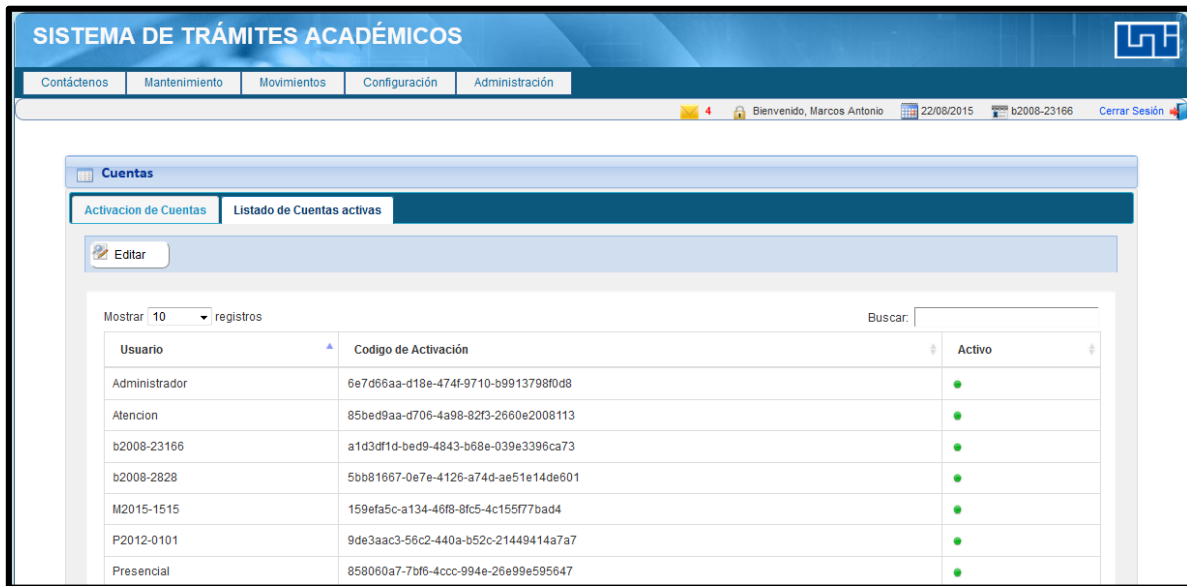
Nombre: Gloria.Thalia

Codigo: 07d30c05-2e92-4ab2-bfd8-e50befda129f

Activo: ☒ Activo

Regresar Actualizar

- Presionar el botón **“Actualizar”** para cambiar el estado de la cuenta.
- Se re direccionará al formulario anterior y las cuentas inactivas ya no se mostraran.



Usuario	Codigo de Activación	Activo
Administrador	6e7d66aa-d18e-474f-9710-b9913798f0d8	●
Atencion	85bed9aa-d706-4a98-82f3-2660e2008113	●
b2008-23166	a1d3df1d-bed9-4843-b68e-039e3396ca73	●
b2008-2828	5bb81667-0e7e-4126-a74d-ae51e14de601	●
M2015-1515	159efa5c-a134-46f8-8fc5-4c155f77bad4	●
P2012-0101	9de3aac3-56c2-440a-b52c-21449414a7a7	●
Presencial	858060a7-7bf6-4ccc-994e-26e99e595647	●

- La otra pestaña muestra la lista de las cuentas activas por usuario.
- Para desactivar una cuenta se tiene que **seleccionar** la fila que contiene el registro de la cuenta que se requiere activar.
- Después de seleccionar el registro, presionar el botón **“Editar”**.
- Se cargara contenido del registro y el campo **activo** (activo).
- Cambiar el estado del campo **activo a** (desactivo).

 Activo'. There are 'Regresar' and 'Actualizar' buttons at the bottom." data-bbox="137 552 864 793"/>

Nombre: Administrador

Codigo: 6e7d66aa-d18e-474f-9710-b9913798f0d8

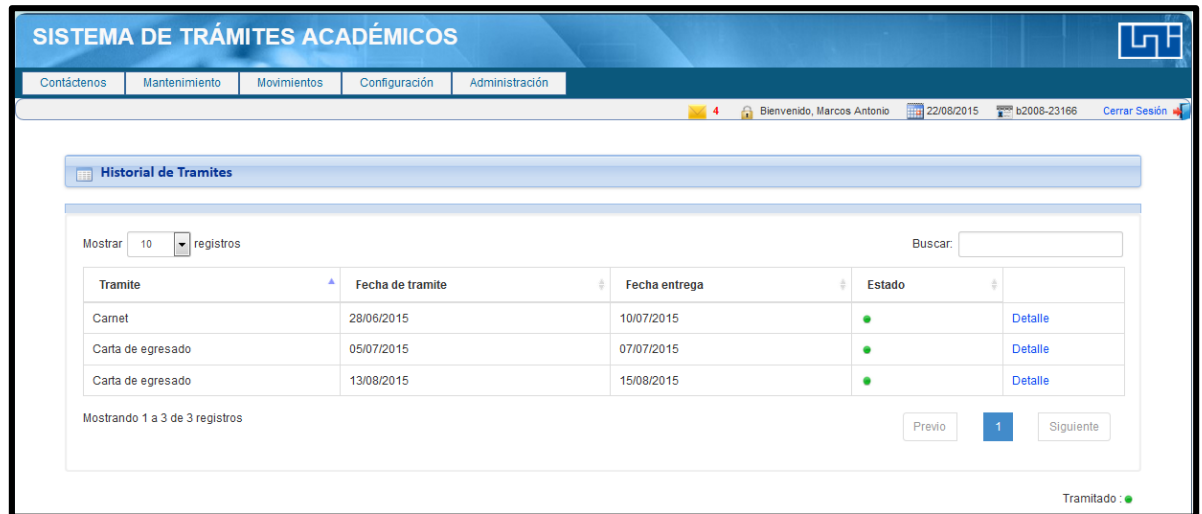
Activo: ☒ Activo

[Regresar](#) [Actualizar](#)

- Presionar el botón **“Actualizar”** para cambiar el estado de la cuenta.
- Esta cuenta desactiva se mostrara en la pestaña primera, donde se muestran las cuentas desactivadas.

## 12. HISTORIAL DE TRÁMITES

En este formulario se mostraran la lista de todos los trámites que se han realizado y que están concluidos.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Historial de Trámites**

Mostrar 10 registros

Buscar:

Trámite	Fecha de trámite	Fecha entrega	Estado	
Carnet	28/06/2015	10/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	05/07/2015	07/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	13/08/2015	15/08/2015	●	<a href="#">Detalle</a>

Mostrando 1 a 3 de 3 registros

Previo 1 Siguiente

Tramitado: ●

- En cada fila que contiene un registro, existe una opción **“Ver Detalle”** en esta opción se mostrara información importante del solicitante y del trámite.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Historial de Trámites**

Mostrar 10 registros

Buscar:

Trámite	Fecha de trámite	Fecha entrega	Estado	
Carnet	28/06/2015	10/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	05/07/2015	07/07/2015	●	<a href="#">Detalle</a>
Carta de egresado	13/08/2015	15/08/2015	●	<a href="#">Detalle</a>

Mostrando 1 a 3 de 3 registros

Previo 1 Siguiente

Tramitado: ●

**Detalle de Trámite**

Tramitante: Jordi Manuel Picado Saldivar

Facultad: FEC

Carrera: Ing. Computacion

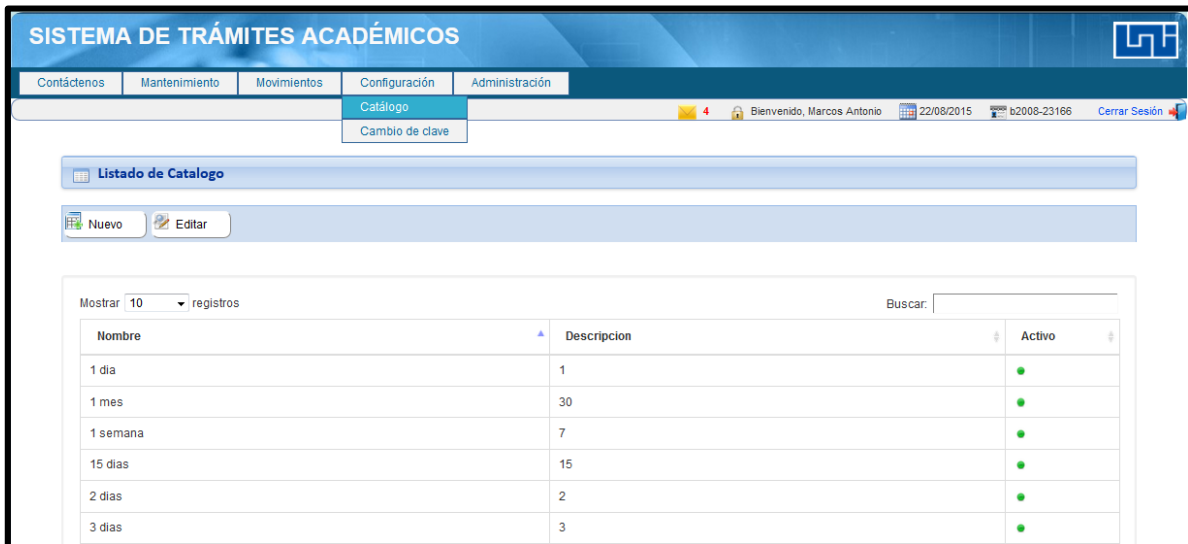
Voucher: 125

Estado: Autorizado

[Aceptar](#)

## 13. CATALOGO

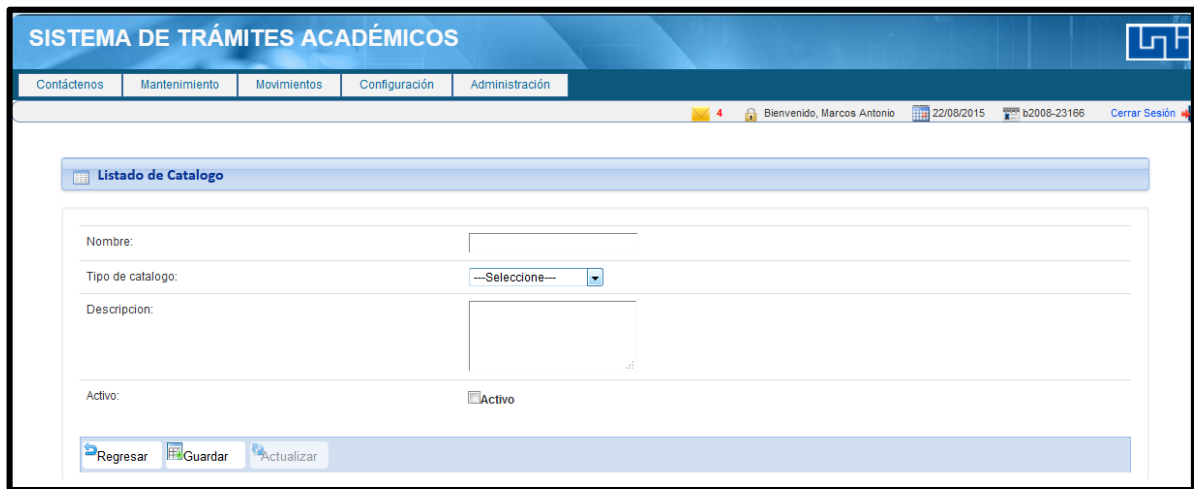
En este formulario se crearan los catálogos del sistema web como por ejemplo (Tipos de documentos, duración, tipo de tarjeta etc.).



Nombre	Descripción	Activo
1 día	1	●
1 mes	30	●
1 semana	7	●
15 días	15	●
2 días	2	●
3 días	3	●

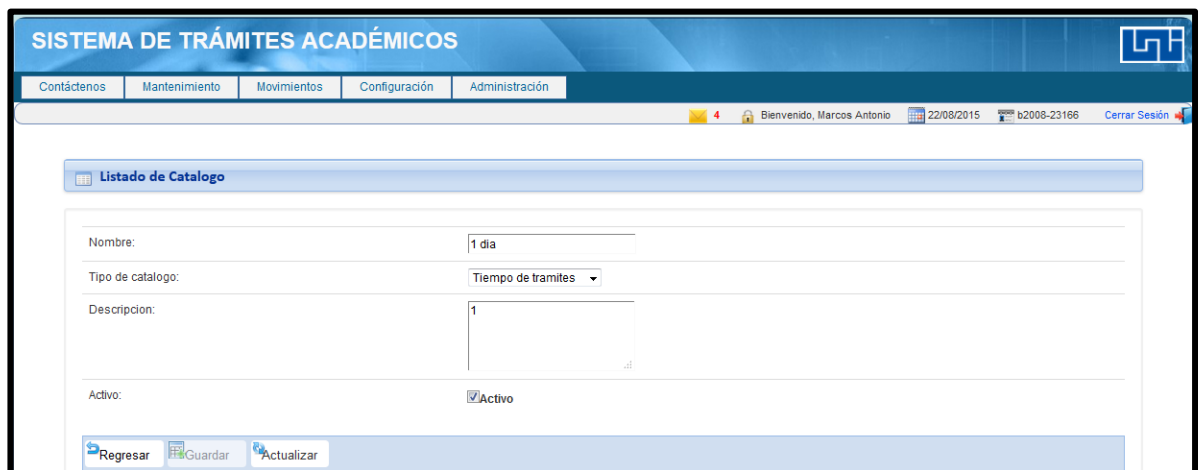
- Primeramente dirigirse a la opción del menú “**Configuración**”.
- Luego elegir la opción “**Catalogo**”.
- Para crear un nuevo catálogo, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.





### 13.1 Edición de Catálogos

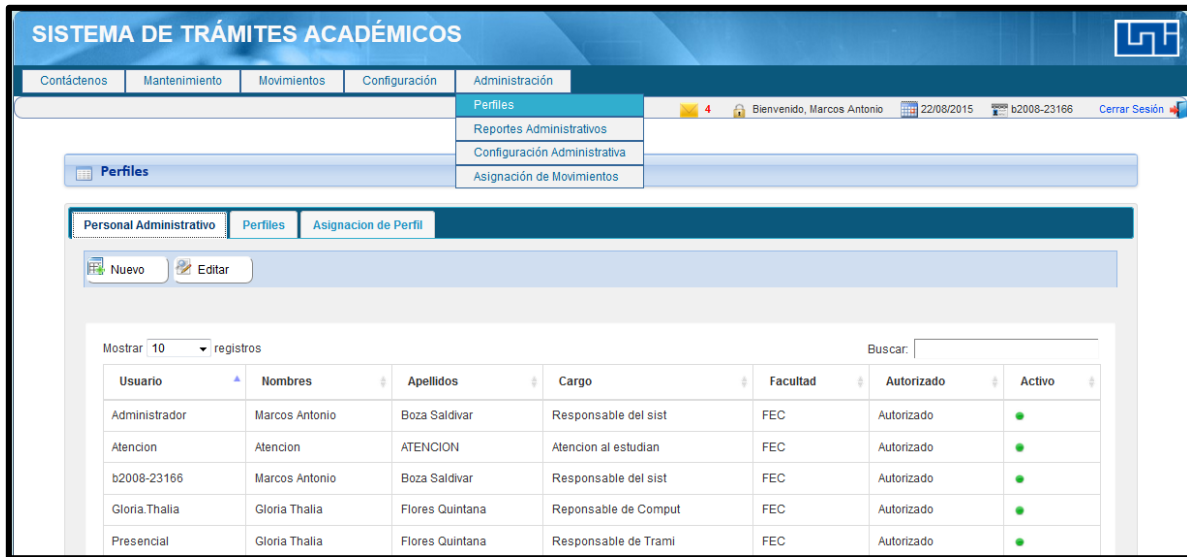
- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 14.ADMINISTRACIÓN

En este grupo de opciones se encuentran las alternativas que tienen acceso el personal administrativo que van desde la creación de perfiles, personal y asignación.



Usuario	Nombres	Apellidos	Cargo	Facultad	Autorizado	Activo
Administrador	Marcos Antonio	Boza Saldivar	Responsable del sist	FEC	Autorizado	●
Atencion	Atencion	ATENCION	Atencion al estudian	FEC	Autorizado	●
b2008-23166	Marcos Antonio	Boza Saldivar	Responsable del sist	FEC	Autorizado	●
Gloria.Thalia	Gloria Thalia	Flores Quintana	Reponsable de Comput	FEC	Autorizado	●
Presencial	Gloria Thalia	Flores Quintana	Responsable de Trami	FEC	Autorizado	●

### 14.1 Perfiles

En este formulario se crean los perfiles que estarán disponibles para el sistema y que serán asignados al personal administrativo y estudiantes, en dependencia de estos perfiles estarán los privilegios de acceso.



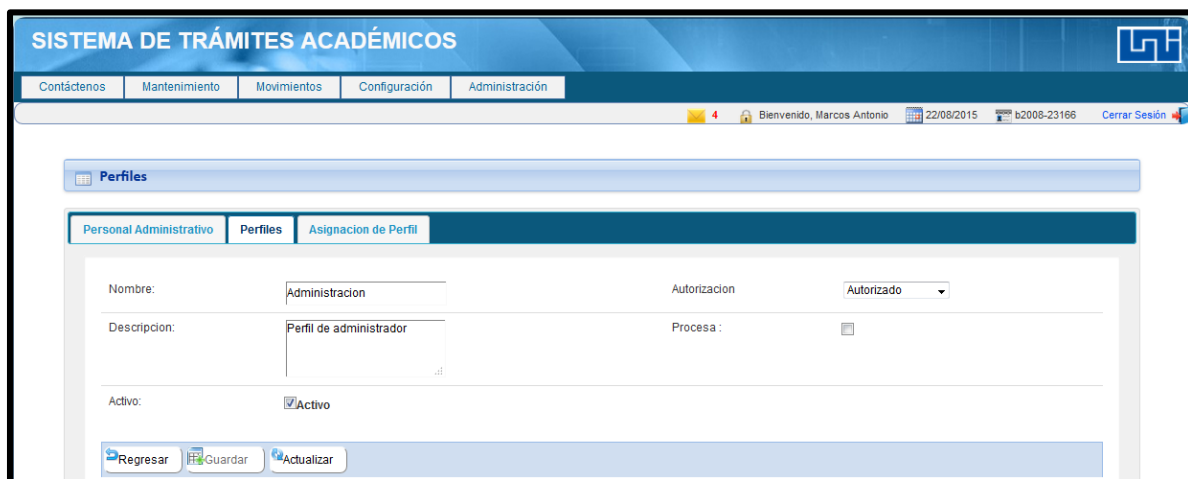
Perfil	Descripción	Autorizacion	Activo
Administracion	Perfil de administrador	Autorizado	●
Atencion	Perfil para usuario no registrado	Autorizado	●
Encargad@ Computacion	Se encargara de recibir las solicitudes de los estudiantes de la carrera de computacion	Autorizado	●
Encargad@ de Electrica	Encargad@ de la bandeja de las solicitudes de la carrera de Electrica	Autorizado	●
Estudiante	Perfil para estudiantes	Autorizado	●

- Para crear un nuevo perfil, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- ✓ El campo procesar indica que el perfil tendrá privilegios para procesar tramites académicos.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.



## 14.2 Edición de perfil

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

### 14.3 Personal Administrativo

Son los usuarios que tienen acceso especial al sistema web, por ejemplo encargados de carreras en la secretaria de facultad o administradores.



- Primeramente dirigirse a la opción del menú “**Administración**”.
- Luego elegir la pestaña “**Personal administrativo**”.
- Para crear un nuevo catálogo, presionar el botón **Nuevo**.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.

### 14.4 Edición de personal administrativo

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Perfiles**

Personal Administrativo | **Perfiles** | Asignación de Perfil

Usuario: Administrador  
 Nombres: Marcos Antonio  
 Apellidos: Boza Saldivar  
 Cargo: Responsable del sist  
 Correo:   
 Clave:   
 Facultad: FEC  
 Activo: ☒ Activo  
 Estado: Autorizado

Regresar | Guardar | Actualizar

- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

## 14.5 Asignación de Perfiles

En el formulario de asignación de perfiles se mostrará todo el personal administrativo registrado en el sistema y en cada fila de registros una opción para editar el perfil asignado.



**SISTEMA DE TRÁMITES ACADÉMICOS**

Contáctenos | Mantenimiento | Movimientos | Configuración | Administración

Bienvenido, Marcos Antonio | 22/08/2015 | b2008-23166 | Cerrar Sesión

**Perfiles**

Personal Administrativo | Perfiles | **Asignación de Perfil**

Usuario	Perfil	
b2008-23166	Administracion	
Administrador	Encargad@ Computacion	
Reynalbarra15	Atencion	
Presencial	Atencion	
Gloria.Thalia	Encargad@ Computacion	
Atencion	Atencion	

## 14.6 Edición de asignación de perfil



Usuario	Perfil	
b2008-23166	Administración	 
Administrador	Estudiante	
Reynalbarra15	Atención	
Presencial	Encargad@ Computación	
Gloria.Thalia	No definido	
Atención	Encargad@ de Eléctrica	

- Para la edición solo basta con presionar el botón de **editar** ubicado al lado derecho de cada registro en el formulario.
- Se mostrara una lista que contiene los perfiles disponibles.
- Luego de elegir el perfil, presionar el botón de **actualizar**, de lo contrario presionar el botón **cancelar**.

## 15.REPORTES ADMINISTRATIVOS

Los reportes administrativos son un conjunto de reportes que presentan la información más importante y de forma generalizada de todos los trámites académicos, solicitudes, accesos, transacciones.



Acceso de Usuarios | Solicitud de cambios de clave | Historial de tramites | Transacciones | Porcentaje de Tramites

Fecha Inicio :

Fecha Fin:



- Para acceder a los reportes dirigirse a la opción “**Administración**”.

- Se desplegarán una lista de opciones, elegir la opción “**Reportes Administrativos**”

### 15.1 Acceso de Usuarios

En este reportes se mostraran los usuarios que han tenido acceso al sistema, su fecha de ingreso y la última conexión.



- Para generar el reporte hay que ingresar el rango de fecha, una fecha de inicio y una fecha fin.
- Presionar el botón “**Generar**”.

1 of 1 Find   Next		
		
SISTEMA DE TRAMITES ACADEMICOS		
Fecha Impresion: 22/08/2015 23:28:29		
Usuario	Fecha de creacion	Ultima coneccion
P2012-0101	28/06/2015 03:30	07/07/2015 02:31 p.m.
b2008-2828	05/07/2015 09:20	26/07/2015 04:45 p.m.
M2015-1515	13/08/2015 02:17	13/08/2015 02:17 p.m.
b2008-23166	30/10/2014 11:04	22/08/2015 09:55 p.m.
Administrador	28/06/2015 03:24	07/07/2015 09:23 a.m.
Reynalbarra15	01/07/2015 10:24	27/07/2015 11:06 p.m.
Presencial	13/08/2015 01:26	13/08/2015 01:35 p.m.
Gloria.Thalia	13/08/2015 02:25	13/08/2015 03:55 p.m.
Atencion	13/08/2015 02:28	13/08/2015 04:00 p.m.

## 15.2 Cambio de clave

Este reporte mostrará los registros de todos los cambios de claves que han realizado los usuarios, todo esto para mantener un control de estas acciones.



- Para generar el reporte hay que ingresar el rango de fecha, una fecha de inicio y una fecha fin.
- Presionar el botón **“Generar”**.



Usuario	Nombres	Apellidos	Recinto	Facultad	Creacion	Fecha Cambio
Ing. Computacion						
P2012-0101	Jordi Manuel	Picado Saldivar	RUSB	FEC	28/06/2015 15:30:35	07/07/2015 02:32:10 p.m.

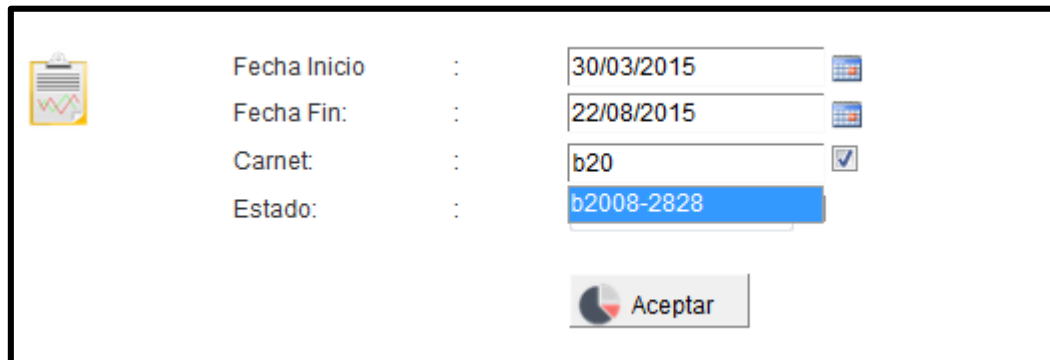


### 15.3 Historial de Tramites

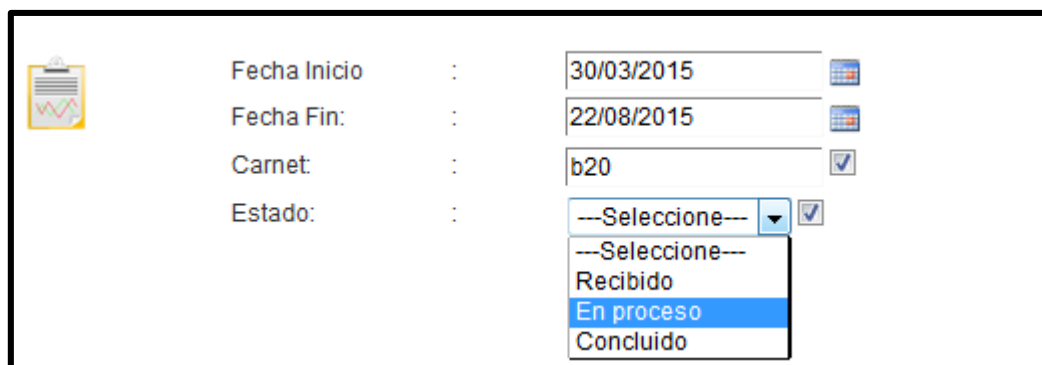
Este reporte mostrara los registros de todas las solicitudes de trámites independientemente del estado en que se encuentren.



- Para generar el reporte hay que ingresar el rango de fecha, una fecha de inicio y una fecha fin.
- Puede filtrar por el número de carnet específico.



- Puede filtrar por estado de trámites.



- Presionar el botón “**Generar**”.

1 of 1 Find   Next				
				
SISTEMA DE TRAMITES ACADEMICOS				
Fecha Impresion: 22/08/2015 23:41:06				
Registro	Carnet	Fecha Solicitud	Fecha Entrega	Estado
Carnet				
1	P2012-0101	28/06/2015	10/07/2015	Concluido
Carta de egresado				
2	b2008-2828	05/07/2015	07/07/2015	Concluido
3	b2008-2828	07/07/2015	09/07/2015	En proceso
4	P2012-0101	07/07/2015	09/07/2015	En proceso
5	b2008-2828	24/07/2015	26/07/2015	Recibido
6	M2015-1515	13/08/2015	15/08/2015	Concluido
7	P2012-0101	13/08/2015	15/08/2015	Recibido

## 15.4 Reporte de Transacciones

Este reporte mostrará el historial de transacciones realizadas independiente del estado y tipo de transacción.

SISTEMA DE TRÁMITES ACADÉMICOS

Contáctenos
Mantenimiento
Movimientos
Configuración
Administración

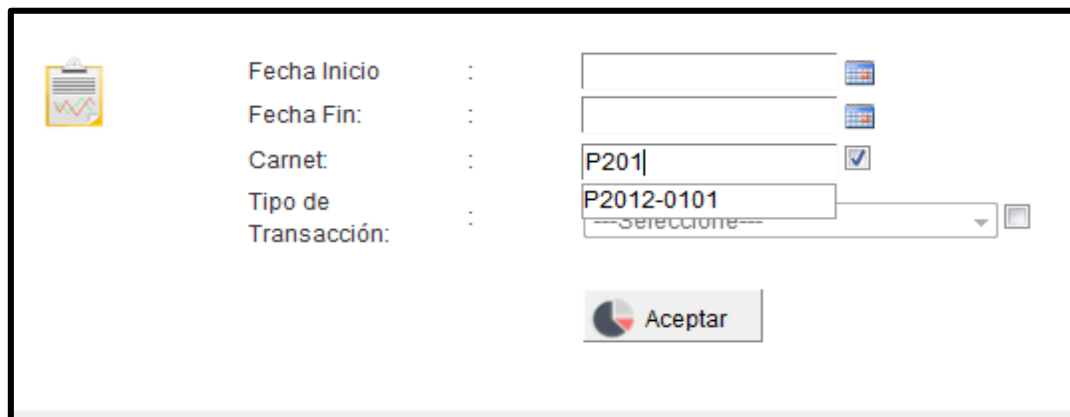
4
Bienvenido, Marcos Antonio
22/08/2015
b2008-23166
Cerrar Sesión

Acceso de Usuarios
Solicitud de cambios de clave
Historial de tramites
**Transacciones**
Porcentaje de Tramites



Fecha Inicio : 
Fecha Fin: 
Carnet: 
Tipo de Transacción:

- Para generar el reporte hay que ingresar el rango de fecha, una fecha de inicio y una fecha fin.
- Filtrar por el número de carnet específico.



Fecha Inicio :

Fecha Fin: :

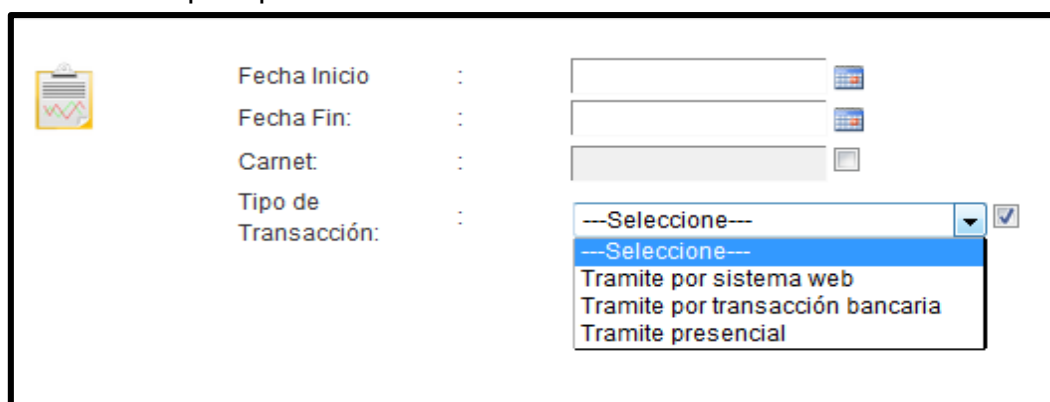
Carnet: : P201

Tipo de Transacción: : P2012-0101

Transacción: : --Seleccione--

Aceptar

- Puede filtrar por tipo de transacción.



Fecha Inicio :

Fecha Fin: :

Carnet: :

Tipo de Transacción: : --Seleccione--

Transacción: : --Seleccione--  
Tramite por sistema web  
Tramite por transacción bancaria  
Tramite presencial

- Presionar el botón “Generar”.

1 of 1 Find | Next

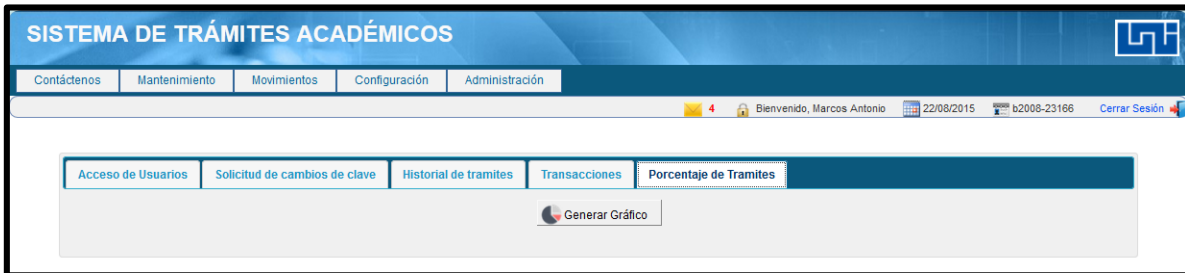
**SISTEMA DE TRAMITES ACADEMICOS**

Fecha Impresion: 22/08/2015 23:48:20

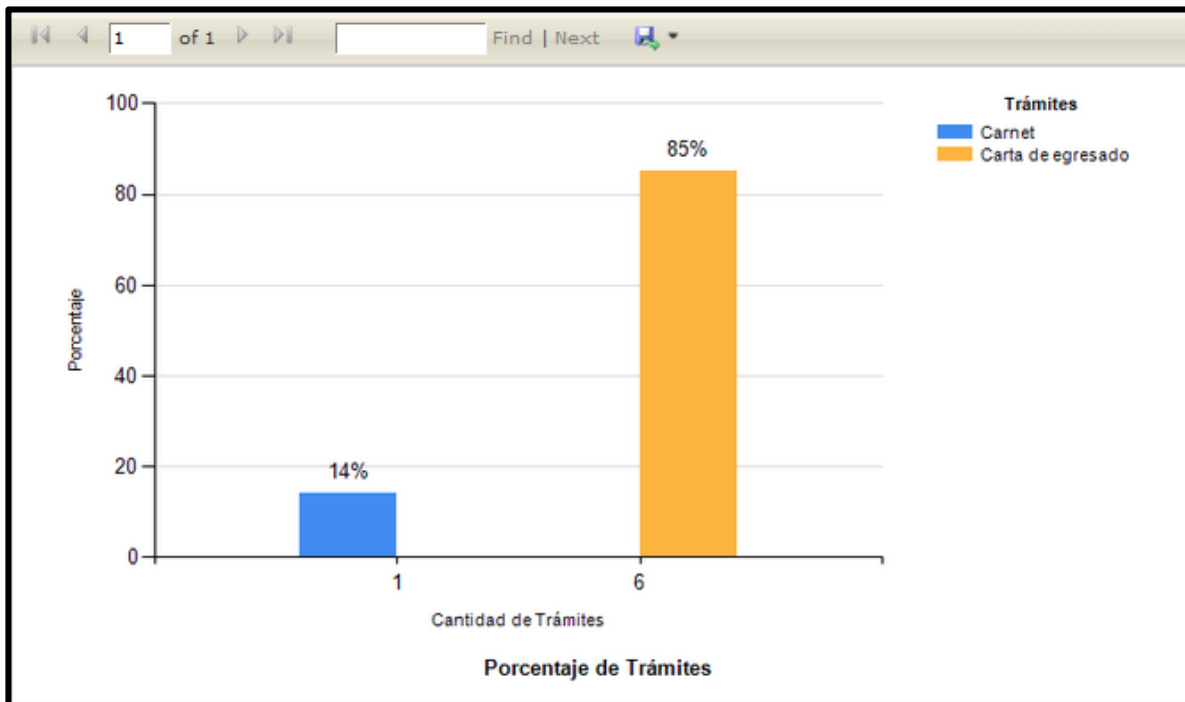
Carnet	Voucher	Fecha	Monto	Estado
<b>Banpro</b>				
b2008-2828	123456	24/07/2015	50	Autorizado
<b>Sistema web</b>				
P2012-0101	125	28/06/2015	770	Autorizado
b2008-2828	126	05/07/2015	75	Autorizado
P2012-0101	128	07/07/2015	75	Autorizado
M2015-1515	130	13/08/2015	75	Autorizado
<b>Tesoreria</b>				
b2008-2828	78975	07/07/2015	75	Autorizado
P2012-0101	78975	13/08/2015	75	Autorizado

## 15.5 Reporte grafico de porcentaje de trámites

Este reporte mostrará gráficamente el porcentaje que representa cada solicitud de trámite realizada ante el total de trámites realizados.

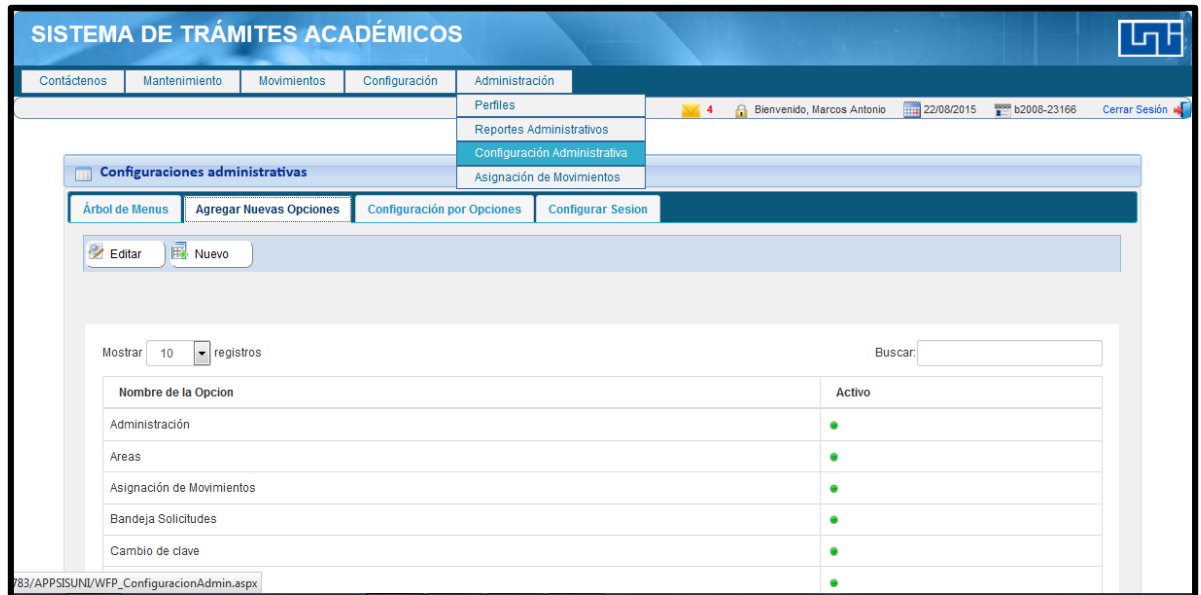


- Para generar el reporte solo se debe presionar el botón “**Generar**”.



## 16. CONFIGURACIÓN ADMINISTRATIVA

Esta opción contiene la funcionalidad de agregar dinámicamente nuevas opciones en el menú, indicar que perfiles pueden ver las opciones y configurar el tiempo de sesión para los usuarios.



- Para ingresar a estas opciones, tiene que dirigirse a la opción “Administración”.
- Luego elegir la opción “Configuración Administrativa”

### 16.1 Agregar nuevas Opciones



- Para crear una nueva opción, presionar el botón **Nuevo**.
  - ✓ El campo Tipo: indica si la opción será de raíz en el menú principal o será hijo de alguna otra opción.
  - ✓ En caso de que sea de Raíz en el campo URL deberá Ingresar el símbolo “#”.
  - ✓ De lo contrario deberá poner la URL del Archivo de código de la nueva opción.
- Se cargara un formulario para rellenar los campos requeridos, al terminar presionar el botón **Guardar**.
- Si se guardó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.

## 16.2 Edición de Nueva Opción

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.



The screenshot displays the 'SISTEMA DE TRÁMITES ACADÉMICOS' interface. The top navigation bar includes links for 'Contáctenos', 'Mantenimiento', 'Movimientos', 'Configuración', and 'Administración'. The 'Configuración' section is active, showing a sub-menu with 'Árbol de Menus', 'Agregar Nuevas Opciones', 'Configuración por Opciones', and 'Configurar Sesión'. The 'Configuración por Opciones' form is visible, containing the following fields and controls:

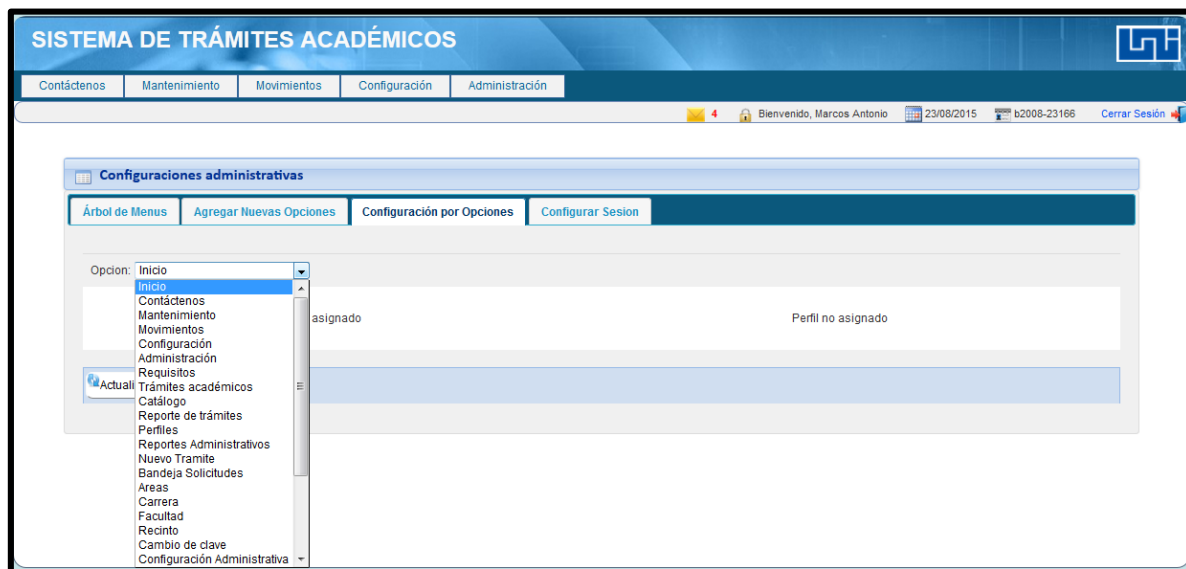
- Nombre:** A text input field containing 'Administración'.
- Tipo:** A dropdown menu set to 'Padre'.
- Url:** A text input field containing '#', with a small 'id' label to its right.
- Padre:** A dropdown menu set to 'Administración'.
- Activo:** A checkbox labeled 'Activo' which is checked.

At the bottom of the form, there are three buttons: 'Regresar', 'Actualizar', and 'Guardar'.

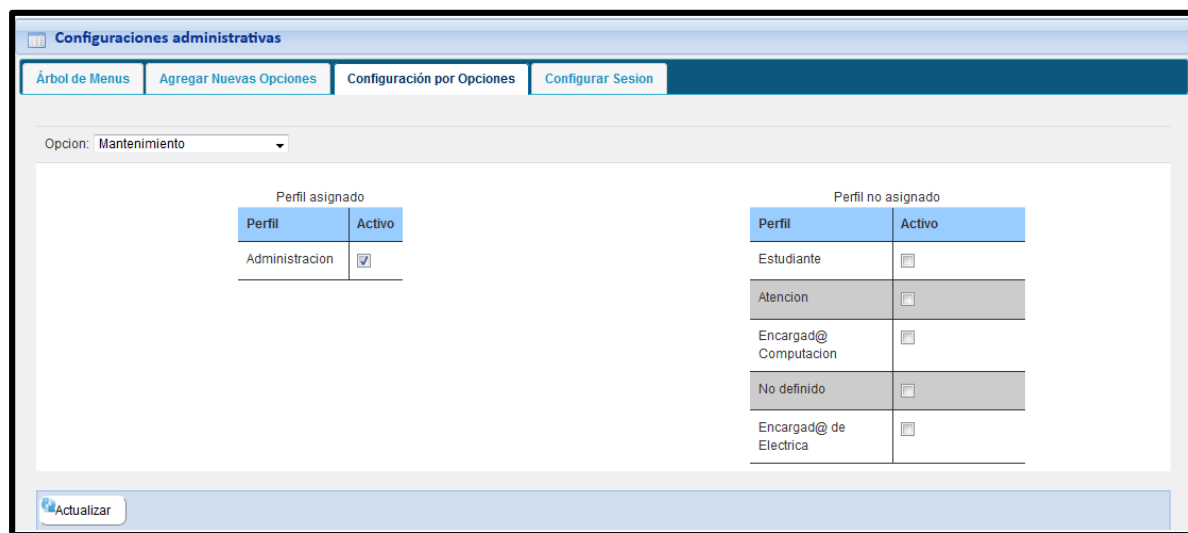
- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.

### 16.3 Configuración de Opciones por Perfil

En esta sección se podrá elegir el nivel de privilegios indicando que opciones se podrán visualizar por perfil en el sistema Web.



- Para configurar que opciones estarán disponibles por cada tipo de Perfil, primero seleccione una opción del control que dice “**Opción**”.
- Luego se cargaran en 2 tablas los perfiles que pueden visualizar las opciones a la izquierda y los que no a la derecha.
- Póngales un check los perfiles que requiere que se les muestre las opción y quite el check a los que no.





Movimiento	Recinto	Facultad	Carrera	Activo
Bandeja Computacion	RUSB	FEC	Ing. Computacion	●
Bandeja Electrica	RUPAP	FEC	Ing. Electrica	●

- Para guardar los cambios seleccione el botón “Actualizar”.
- Si se actualizó el registro se mostrara una notificación exitosa, de lo contrario se mostrara una notificación indicando un error.

## 16.4 Configuración de Sesión

En esta opción, el usuario determina el tiempo que se prolongara la sesión de los usuarios independientemente del perfil.



- Simplemente elija el tiempo a transcurrir, ya sea (1 hr, 6 hrs, 12 hrs o 1 día).
- Luego presionar el botón “**Actualizar**”

## 17. MOVIMIENTOS

Aquí se crean los movimientos que serán asignados a los perfiles, por ejemplo se sabe que en la secretaria de la facultad de electrotecnia y computación (FEC) se encuentran un encargado por cada carrera entonces a cada encargado se le asignara un movimiento diferente que apuntan a un recinto, a una facultad y a una carrera en específicos.



- Se deben ingresar los datos requeridos, de lo contrario aparecerá un mensaje notificando que hay campos vacíos.
- Después de llenar los campos correctamente, presionar el botón **Guardar** que devolverá al formulario anterior mostrando el registro que se ingresó y una notificación indicando que el registro fue almacenado.

### 17.1 Edición de Requisitos

- Para la edición de registros, se deben de seleccionar el registro a modificar, luego presionar el botón **Editar**.

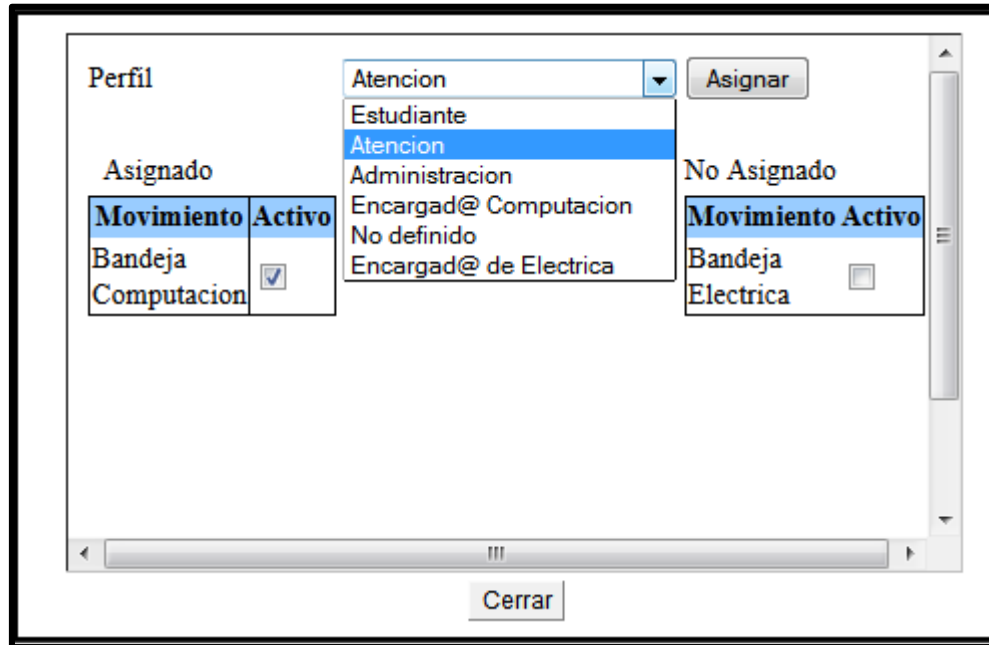


The screenshot displays the 'SISTEMA DE TRÁMITES ACADÉMICOS' web interface. At the top, there is a navigation bar with tabs: 'Contáctenos', 'Mantenimiento', 'Movimientos', 'Configuración', and 'Administración'. Below this, a user status bar shows 'Bienvenido, Marcos Antonio', the date '23/08/2015', and a session ID 'b2008-23166'. The main content area is titled 'Asignaciones'. It contains a form with the following fields:

- Movimiento:** A text input field containing 'Bandeja Computacion'.
- Recinto:** A dropdown menu with 'RUSB' selected.
- Descripción:** A text input field containing 'Solicitudes de los estudiante de computacion'.
- Facultad:** A dropdown menu with 'FEC' selected.
- Activo:** A checkbox labeled 'Activo' which is checked.
- Carrera:** A dropdown menu with 'Ing. Computacion' selected.
- Estado:** A dropdown menu with 'Autorizado' selected.

At the bottom of the form, there are three buttons: 'Regresar', 'Guardar', and 'Actualizar'.

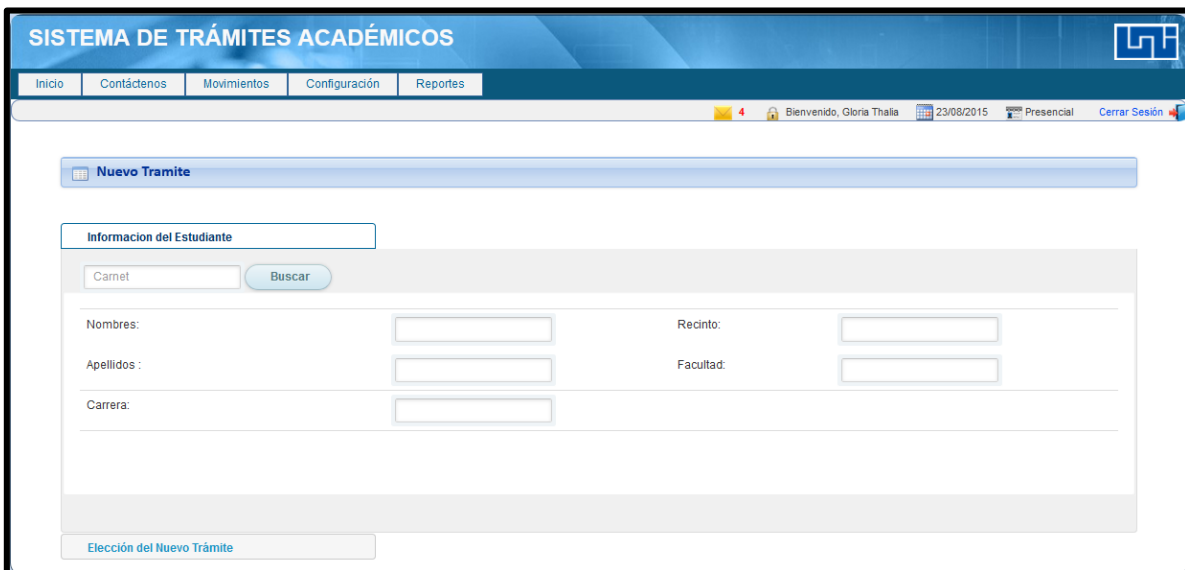
- Se cargara el formulario con los datos del registro a editar y abajo los botones **Regresar** y **Actualizar**.
- Luego de terminar la edición presionar el botón **Actualizar**.
- Para asignar movimientos a los perfiles, debe presionar el botón **“Asignar movimiento”**.
- Se desplegara un formulario emergente que cargara los perfiles y en dos tablas el estado del movimiento si está asignado o no.



- Para guardar los cambios deberá presionar el botón **“Asignar”**.

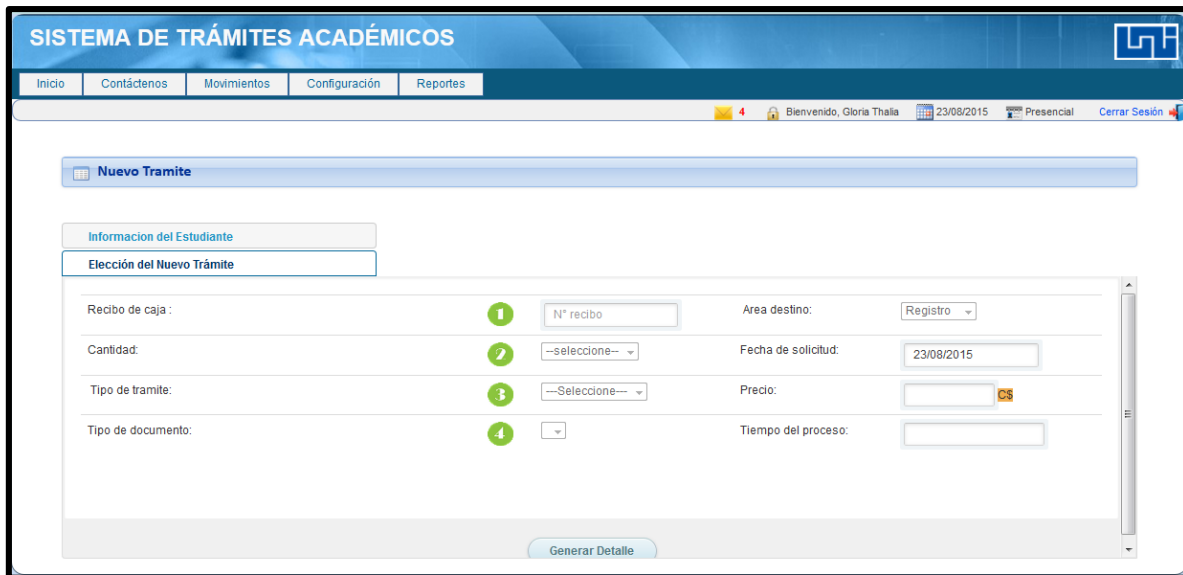
## 18. TRAMITE DIRECTO

Este formulario está diseñado exclusivamente para el personal que reciba las solicitudes de manera presencial.



- Primeramente se mostrara la primera etapa **“Información del estudiante”** que consiste en ingresar el número de carnet del estudiante.

- Luego presionar el botón “**Buscar**”.
- Esperar a que se carguen los datos de verificación del estudiante.
- Luego presionar el nombre de la otra etapa “**Elección del nuevo trámite**”



- Primeramente se tiene que ingresar el número del recibo de factura de pago en tesorería.
- Si el número de recibo es correcto se activaran los siguientes campos que se deben de llenar, de lo contrario no se podrá realizar la solicitud de trámite.



- Luego de llenar los campos requeridos, presionar el botón **“Generar Detalle”**.

Confirmación

Detalle de Transacción

**Certificado de Pago**

---

Tramitante:	Marcos Antonio Boza Tucler
Carnet:	b2008-2828
Concepto:	
	Carta de egresado
Precio:	C\$ 75
Cantidad:	1


---

Fecha de Solicitud :	23/08/2015
Fecha de Entrega :	25/08/2015

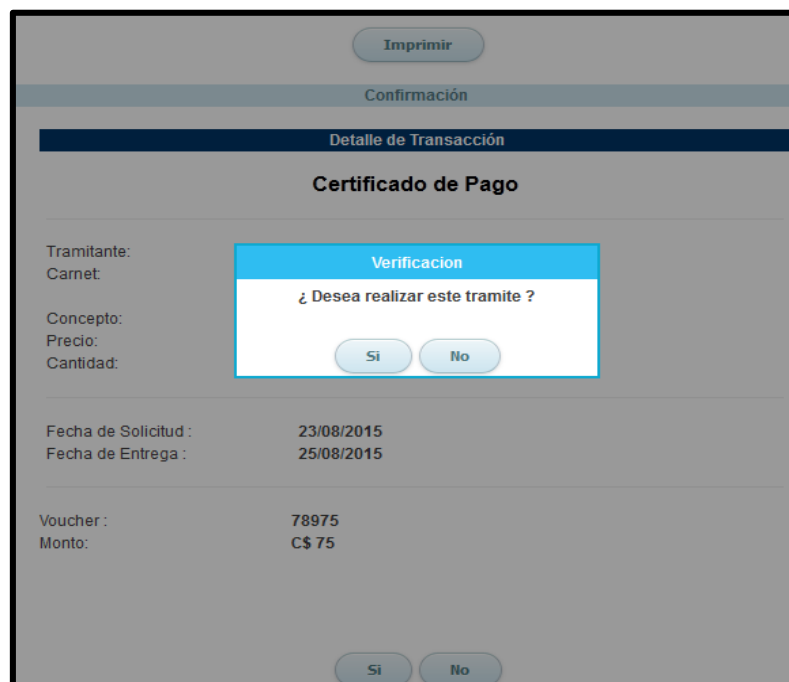
---

Voucher :	78975
Monto:	C\$ 75

- Si desea imprimir el detalle presione el botón **“Imprimir”**. Se mostrara un vista miniatura del detalle que se está realizando.

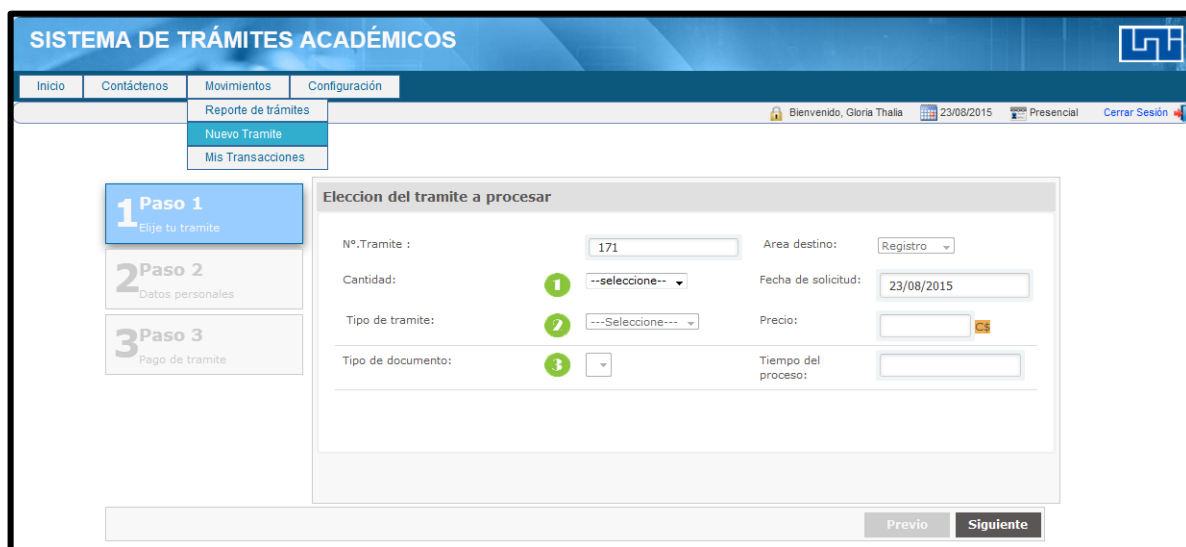
Detalle de Transacción	
<b>Certificado de Pago</b>	
Tramitante:	<b>Reyna Elliette Ibarra Lanzas</b>
Carnet:	<b>s2008-2015</b>
Concepto:	<b>Carta de egresado</b>
Precio:	<b>C\$ 75</b>
Cantidad:	<b>1</b>
Fecha de Solicitud :	<b>05/09/2015</b>
Fecha de Entrega :	<b>07/09/2015</b>
Voucher :	<b>171</b>
Monto:	<b>C\$ 75</b>
 * F E C *	

- Si desea realizar la solicitud , presione el botón “**Si**” de lo contrario el botón “**No**”



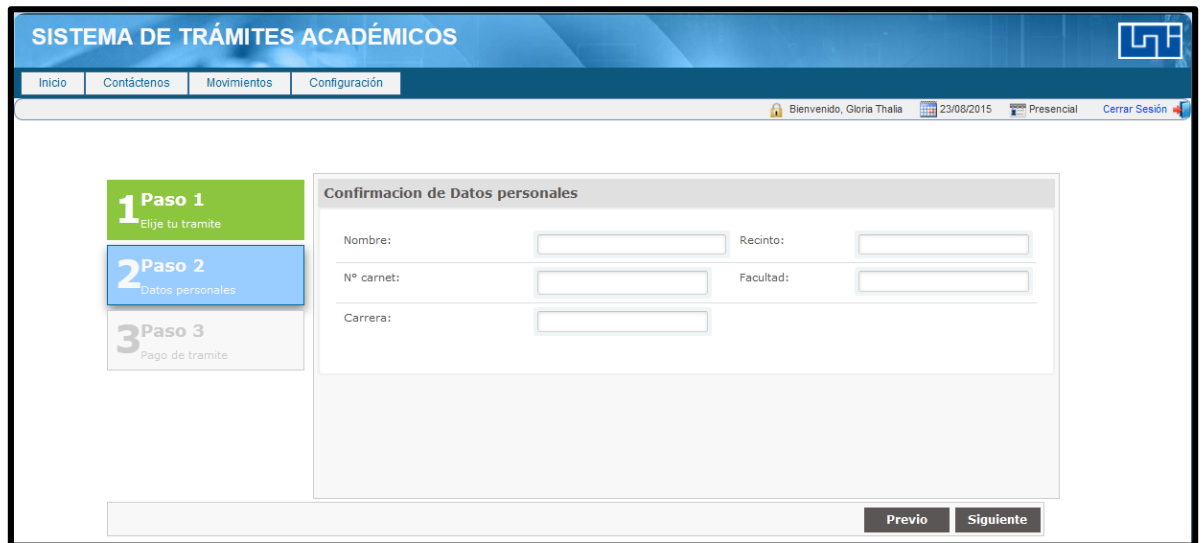
## 19. NUEVO TRAMITE

Este formulario fue creado específicamente para el estudiante, el mismo usuario elegirá su propio tramite, confirmara sus datos personales e ingresar los datos referentes a su tarjeta de crédito o debido para ejecutar dicha solicitud.



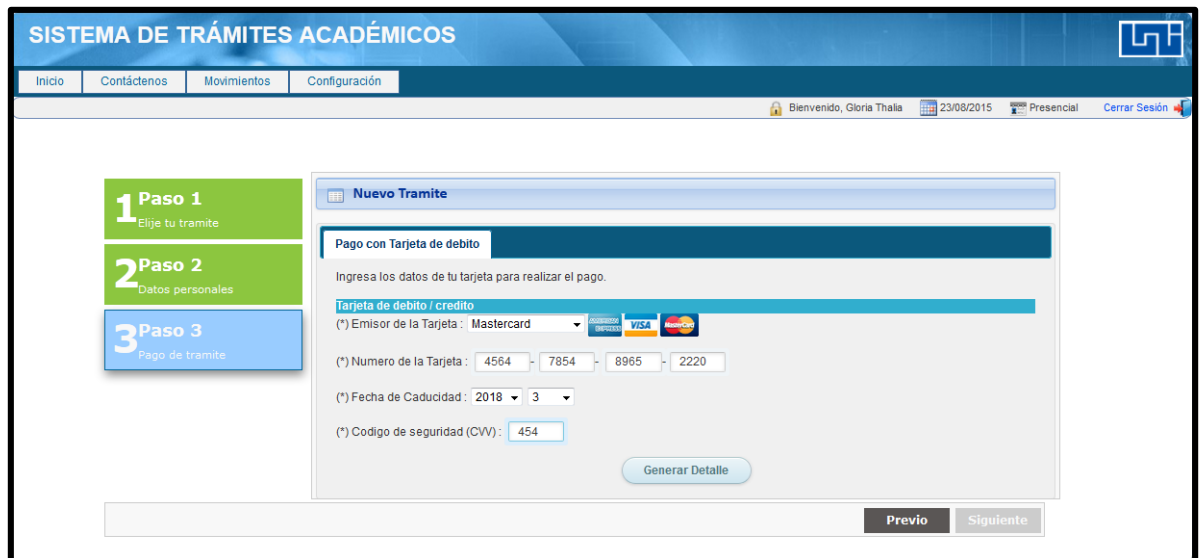
- Para ingresar a estas opciones, tiene que dirigirse a la opción “Movimientos”.

- Luego elegir la opción “**Nuevo trámite**”.
- Después de haber elegido el trámite, presione el botón “**siguiente**”.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' web interface. The top navigation bar includes 'Inicio', 'Contáctenos', 'Movimientos', and 'Configuración'. The user is logged in as 'Bienvenido, Gloria Thalia' on '23/08/2015' with 'Presencial' access. The main content area displays a three-step process on the left: '1 Paso 1 Elije tu tramite' (highlighted in green), '2 Paso 2 Datos personales' (highlighted in blue), and '3 Paso 3 Pago de tramite'. The main form area is titled 'Confirmacion de Datos personales' and contains input fields for 'Nombre:', 'Recinto:', 'N° carnet:', 'Facultad:', and 'Carrera:'. At the bottom right, there are 'Previo' and 'Siguiente' buttons.

- En la segunda etapa el estudiante verifica sus datos estudiantiles.



The screenshot shows the 'SISTEMA DE TRÁMITES ACADÉMICOS' web interface. The top navigation bar is the same. The user is logged in as 'Bienvenido, Gloria Thalia' on '23/08/2015' with 'Presencial' access. The main content area displays the three-step process on the left: '1 Paso 1 Elije tu tramite' (highlighted in green), '2 Paso 2 Datos personales' (highlighted in green), and '3 Paso 3 Pago de tramite' (highlighted in blue). The main form area is titled 'Nuevo Trámite' and shows the 'Pago con Tarjeta de debito' option. It prompts the user to 'Ingresa los datos de tu tarjeta para realizar el pago.' and includes a 'Tarjeta de debito / credito' section with a dropdown for 'Emissor de la Tarjeta' (set to 'Mastercard') and logos for 'American Express', 'VISA', and 'MasterCard'. Below this are input fields for '(\*) Numero de la Tarjeta' (with a masked number '4564 - 7854 - 8965 - 2220'), '(\*) Fecha de Caducidad' (set to '2018' and '3'), and '(\*) Codigo de seguridad (CVV)' (set to '454'). A 'Generar Detalle' button is at the bottom right. At the very bottom, there are 'Previo' and 'Siguiente' buttons.

- En la tercera etapa, el estudiante tiene que ingresar los datos requeridos para poder realizar una compra en línea.
- Luego presionar el botón “**Generar Detalle**”.

**Confirmación**

**Detalle de Transacción**

**Certificado de Pago**

---

Tramitante:	Marcos Antonio Boza Tucler
Carnet:	b2008-2828
Concepto: Carta de egresado	
Precio:	C\$ 75
Cantidad:	1
Fecha de Solicitud : 23/08/2015	
Fecha de Entrega : 25/08/2015	
Voucher : 78975	
Monto:	C\$ 75

- Si desea imprimir el detalle presione el botón “**Imprimir**”. Se mostrara un vista miniatura del detalle que se está realizando.


**Detalle de Transacción**

**Certificado de Pago**

---

Tramitante:	<b>Reyna Elliette Ibarra Lanzas</b>
Carnet:	<b>s2008-2015</b>
Concepto: <b>Carta de egresado</b>	
Precio:	<b>C\$ 75</b>
Cantidad:	<b>1</b>
Fecha de Solicitud : <b>05/09/2015</b>	
Fecha de Entrega : <b>07/09/2015</b>	

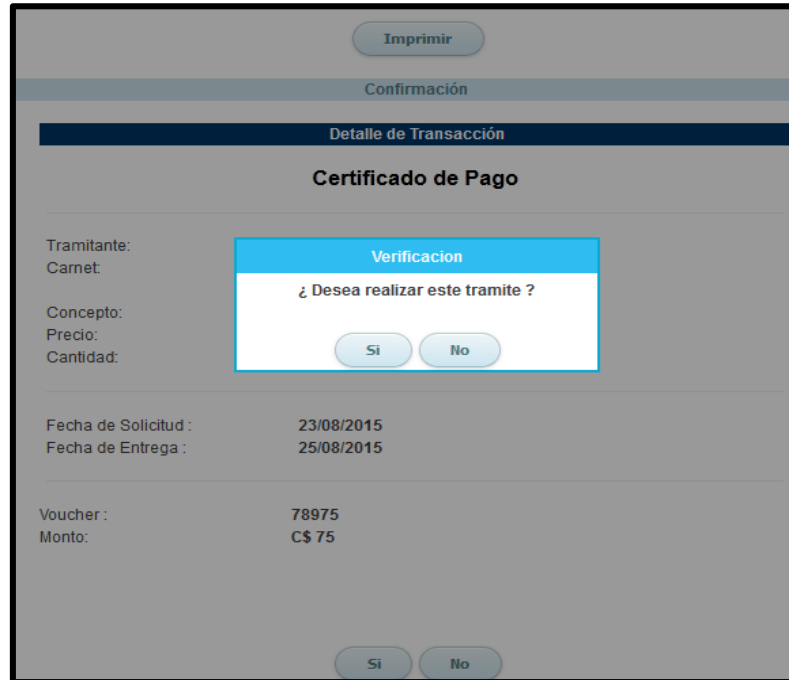
Voucher :	<b>171</b>
Monto:	<b>C\$ 75</b>



★ F E C ★



- Si desea realizar la solicitud , presione el botón “**Si**” de lo contrario el botón “**No**”



The screenshot displays a web interface for transaction confirmation. At the top, there is a button labeled "Imprimir". Below it, a header bar reads "Confirmación". The main section is titled "Detalle de Transacción" and "Certificado de Pago". It contains a form with the following fields:

Tramitante:	
Carnet:	
Concepto:	
Precio:	
Cantidad:	

A modal dialog box titled "Verificacion" is overlaid on the form, asking "¿ Desea realizar este tramite ?" with "Si" and "No" buttons.

Below the form, the following information is displayed:

Fecha de Solicitud :	23/08/2015
Fecha de Entrega :	25/08/2015

At the bottom, the voucher and amount are shown:

Voucher :	78975
Monto:	C\$ 75

At the very bottom, there are "Si" and "No" buttons.

- Si la transacción se realizó correctamente aparecerá una notificación indicando que la solicitud se hizo con éxito, de lo contrario aparecerá una notificación indicando un error.



# Manual Técnico

---

## 20. INTRODUCCIÓN

El siguiente documento describe aspectos técnicos del Sistema web de solicitudes de trámites académicos para la Facultad de Electrotecnia y Computación (FEC) de la Universidad Nacional de Ingeniería (UNI).

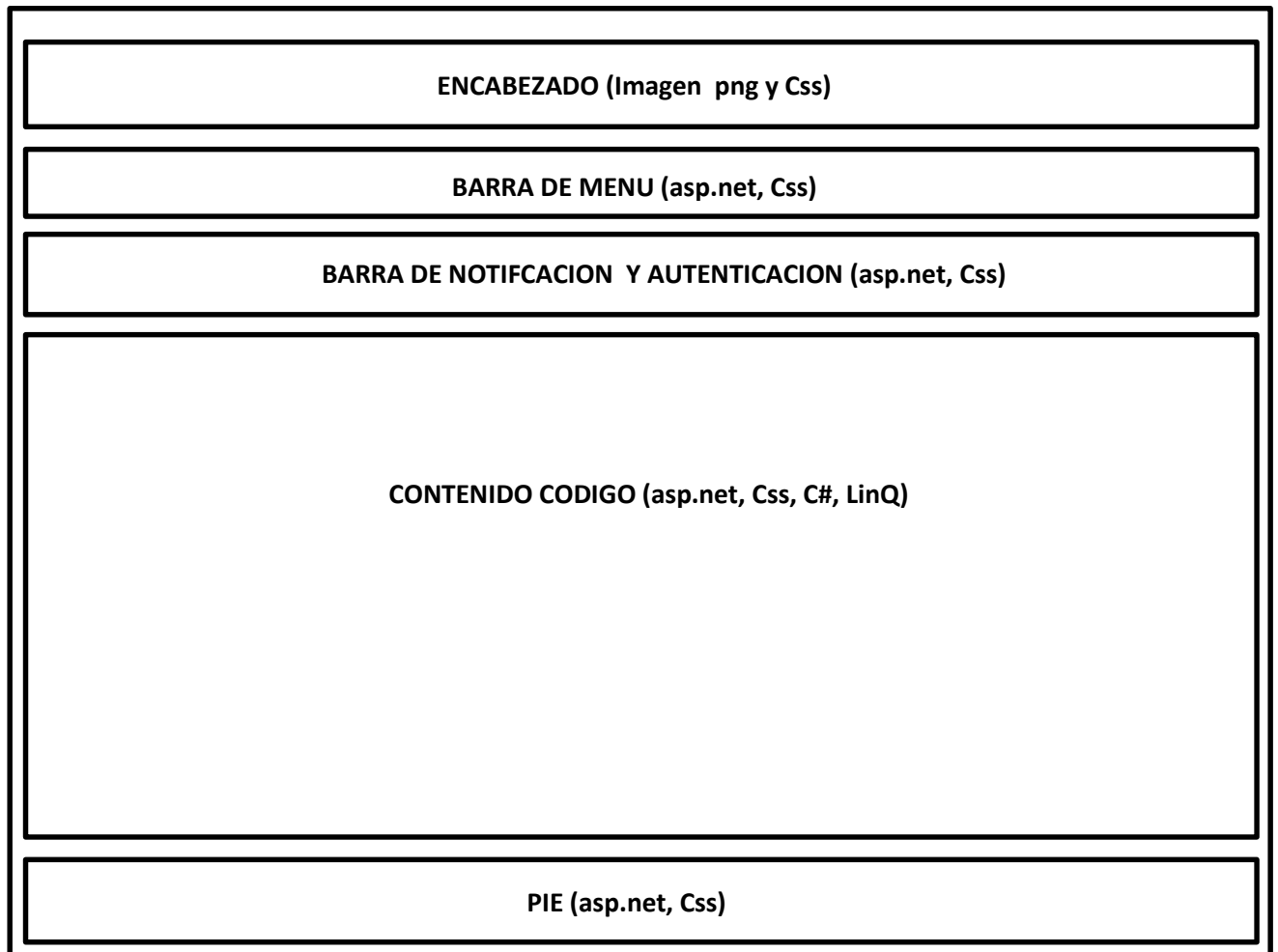
EL sistema web permitirá a los usuarios realizar las solicitudes de trámites académicos, mantener un adecuado seguimiento de los trámites realizados y funciones específicas en dependencia de los niveles de privilegios entre el personal administrativo ayudando a tener un mejor orden y control de las solicitudes y sus estados.

A continuación se detallan generalidades del diseño del sistema, estructura de la base de datos y explicación de códigos fuentes principales dentro de las clases. Los comentarios dentro del código fuente de las clases, explican el uso de cada segmento de código y funciones específicas.

## 21. DISEÑO DE INTERFAZ

Para el diseño de las interfaces se utilizó el Diseñador de Windows Forms integrado en el Visual Studio 2010 generando el código ASP.NET y la Suite de Adobe CS4 para el diseño de los iconos e imágenes.

La interfaz inicial del sistema está conformada por los siguientes elementos:



El diseño de la aplicación es web adaptable o mejor conocido como diseño responsivo cuyo objetivo es adaptar las apariencias de los sistemas web al dispositivo que se esté utilizando.

La interfaz principal que incluye los menús para los diferentes tipos de Usuarios “Estudiantes”, “Personal Administrativo” Y “Administrador” en dependencia de los privilegios asignados al perfil.

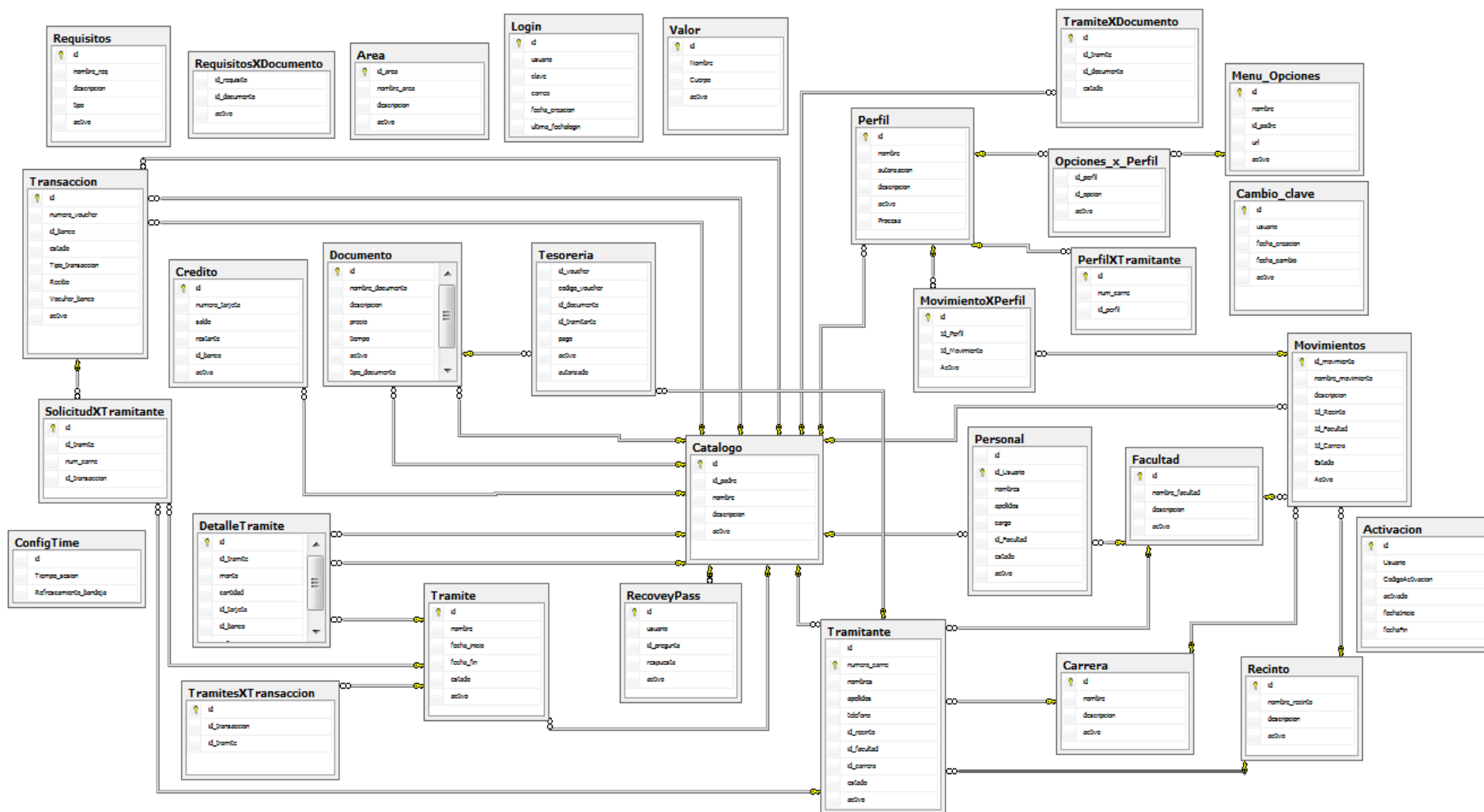
## 22. OPCIONES DE MENU

Las opciones por default que se presentan en el sistema web se encuentran estructuradas de la siguiente forma.

Usuario	Opción de Menú	
Administrador	Inicio	
	Contáctenos	
	Mantenimiento	Requisitos
		Tramites académicos
		Áreas
		Carrera
		Facultad
		Recinto
	Movimientos	Bandeja de Solicitudes
		Cuentas
		Historial de trámites
	Configuración	Catalogo
		Cambio de clave
	Administración	Perfiles
		Reportes Administrativos
		Configuración Administrativa
		Asignación de Movimientos
Personal administración	Inicio	
	Contáctenos	
	Movimientos	Bandeja de trámites
		Trámite Directo

	Configuración	Cambio de Clave
	Reportes	Historial de trámites
Estudiante	Inicio	
	Contáctenos	
		Nuevo trámite
	Movimientos	Mis transacciones
		Reporte de tramites
	Configuración	Cambio de clave

## 23. ESTRUCTURA DE DATOS



## 24. CLASES PRINCIPALES DE LA APLICACIÓN

### 24.1 CONEXIÓN EN EL SISTEMA WEB

#### 24.1.1 Conexión

Cadena de Conexión en el archivo de configuración **Web.config**. El archivo de configuración: en este archivo se especifican los parámetros necesarios para que la aplicación se conecte a la base de datos (usuario, contraseña, base de datos, servidor de base de datos).

```
/** Cadena de conexión con el servidor de Base de datos*/  
  
    <add name="Tramites_AcademicosConnectionString"  
        connectionString="Data Source=.;Initial  
Catalog=Tramites_Academicos;Integrated Security=True"  
        providerName="System.Data.SqlClient" />
```

#### 24.1.2 Clase de Conexión

Esta es la clase principal que comunica el sistema web desarrollado en C# y asp.net con la base de datos en SQL Server haciendo uso del nombre de la cadena de conexión.

```
Public class Conexion  
{  
    Public static SqlConnection Conectar()  
    {  
        Try  
        {  
  
            /** Nombre de la conexión con el servidor de Base de datos*/  
  
            String conn =  
            ConfigurationManager.ConnectionStrings["Tramites_AcademicosConnect  
ionString"].ConnectionString;  
            SqlConnection cn = new SqlConnection(conn);  
            Return cn;  
  
        }  
    }  
}
```



```

        Catch (Exception ex)
        {
            Throw new ArgumentException ("Error al conectar", ex);
        }
    }
}

```

## 24.2 CLASES DE ACCESO

### 24.2.1 Clase de Acceso

Esta función Valida si los datos de acceso están correctos, si el Usuario existe y si es personal está activo en la universidad al igual verifica si el Usuario tiene activa su cuenta en el sistema web, todo este proceso se realiza haciendo el llamado a un (Procedimiento almacenado).

```

protected void ValidateUser()
{
    int userId = 0;

    /*Se abre la Conexion con la Base de datos*/

    Conexion conexion = new Conexion();
    using (SqlConnection con = Conexion.Conectar())
    {
        /* Nombre del procedimiento almacenado con sus parámetros*/

        using (SqlCommand cmd = new SqlCommand("Validate_User"))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("@Username", TextBoxUser.Text);

            /*Convierte la contraseña al formato de Seguridad*/
            cmd.Parameters.AddWithValue("@Password",
                Base64_Encode(TextBoxPass.Text));

            cmd.Connection = con;
            con.Open();
            userId = Convert.ToInt32(cmd.ExecuteScalar());
            con.Close();
        }

        switch (userId)
        {
            case 0:

                MessageBoxSafe.SetMessage(App_Controls_MessageBox.MessageType.Warning,
                    "Ingrese los datos correctamente!");
            }
        }
    }
}

```

```

        TextBoxUser.Text = "";
        TextBoxUser.Text = "";
        TextBoxUser.Focus();
        break;

    case -1:

        MessageBoxSafe.SetMessage(App_Controls_MessageBox.MessageType.Error,
            "Usuario o contraseña Invalidos");
        TextBoxUser.Text = "";
        TextBoxUser.Text = "";
        break;
    case -2:

        MessageBoxSafe.SetMessage(App_Controls_MessageBox.MessageType.Informati
on, "Su cuenta no esta activa, contacte a un administrador!");
        TextBoxUser.Text = "";
        TextBoxUser.Text = "";
        break;
    case 1:

        /* Se cargan los datos principales del Usuario*/

        BindInfoUser();

        /* Se direcciona al la pagina por default*/
        //Response.Redirect("Default.aspx");
        Response.Redirect("Default.aspx");

        break;
    }
}
}

```

### 24.2.2 Clase de Registro

Esta función verifica que la cuenta de estudiante esta activa, que la información proporcionada para la creación de la cuenta no se encuentre repetida y cumplan con el formato de seguridad requerido, esta misma función ingresa los datos para la funcionalidad de recuperación de contraseña y cambio de clave. Este proceso se hace con el llamado a un procedimiento almacenado.

```

protected void RegistrarUsuario()
{
    int userId = 0;

    /* Se abre una Conexion con la base de datos*/
    Conexion conexion = new Conexion();
    using (SqlConnection con = Conexion.Conectar())
    {

        /* Nombre del procedimiento almacenado con sus parámetros*/
    }
}

```

```

using (SqlCommand cmd = new SqlCommand("Insertar_Usuario"))
{
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.Parameters.AddWithValue("@Username", txtNumeroCarnet.Text);
    cmd.Parameters.AddWithValue("@Password",
Base64_Encode(txtpassword.Text));
    cmd.Parameters.AddWithValue("@Email", TextBoxEmail.Text);
    cmd.Parameters.AddWithValue("@question",
Int32.Parse(ddlPassRecovery.SelectedValue));
    cmd.Parameters.AddWithValue("@answer",
Base64_Encode(TextBoxPassrecovey.Text));
    cmd.Connection = con;
    con.Open();
    userId = Convert.ToInt32(cmd.ExecuteScalar());
    con.Close();
}
switch (userId)
{
    case 0:
        //Response.Write("<script LANGUAGE='JavaScript' >alert('no se
pudo ingresar datos')</script>");

        MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Error, "No se pudo
ingresar datos");
        break;

    case -1:
        //Response.Write("<script LANGUAGE='JavaScript' >alert('Usuario
ya existe')</script>");

        MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Information, "Usuario
ya existe");
        break;

    case -2:
        //Response.Write("<script LANGUAGE='JavaScript' >alert('Correo
ya existe')</script>");

        MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Information, "Correo
ya existe");
        break;

    case -3:
        // Response.Write("<script LANGUAGE='JavaScript' >alert('Tu
cuenta de estudiante no esta activa')</script>");

        MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Tu cuenta
de estudiante no esta activa");
        break;

    case -4:
        // Response.Write("<script LANGUAGE='JavaScript' >alert('Error al
realizar operacion')</script>");

        MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Error, "Error al
realizar operación");
        break;
    default:

```

```

        {
            Validar_Usuario();

            break;
        }
    }
}

```

### 24.2.3 Clase de Asignación de Perfil

Esta función verifica que el usuario tenga la cuenta activa de lo contrario crea el registro de activación y le asigna al usuario el perfil de estudiante por default, en este proceso se calcula una proyección de vigencia de la cuenta de 5 años a partir de la fecha de creación de la misma.

```

protected void Validar_Usuario()
{
    int userId = 0;

    /* Se abre una Conexion con la base de datos*/

    Conexion conexion = new Conexion();
    using (SqlConnection con = Conexion.Conectar())
    {

        /* Nombre del procedimiento almacenado con sus parámetros*/

        using (SqlCommand cmd = new SqlCommand("Validacion_Activo"))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("@Username", txtNumeroCarnet.Text);
            cmd.Connection = con;
            con.Open();
            userId = Convert.ToInt32(cmd.ExecuteScalar());
            con.Close();
        }
        switch (userId)
        {
            case 0:
                //Response.Write("<script LANGUAGE='JavaScript' >alert('No se pudo ingresar los datos')</script>");

                MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Error, "No se pudo ingresar los datos");
                break;

            case -1:
                {
                    //Response.Write("<script LANGUAGE='JavaScript' >alert('Usuario ya existe')</script>");
                }
            }
        }
    }
}

```

```

MessageBoxError.SetMessage(App_Controls_MessageBox.MessageType.Information, "Usuario
ya existe");
        break;
    }

    case 1:
    {

        // Response.Write("<script LANGUAGE='JavaScript'
>alert('Usted ya esta registrado')</script>");

        mvProductos.SetActiveView(vgrid);

        MessageBoxSafe.SetMessage(App_Controls_MessageBox.MessageType.Success, "Usted ya
esta registrado, espere a que su cuenta se active");
        break;
    }
}
}
}

```

#### 24.2.4 Clase de Recuperación de Contraseña

Este método hace la recuperación de la contraseña de la cuenta funcionando bajo un sistema de pregunta de seguridad que se dio respuesta al momento de registrarse por primera vez en el sistema web, verifica la pregunta y la respuesta con lo almacenado en la base de datos. Si el proceso es exitoso se enviara un correo con la contraseña al correo registrado con anterioridad.

```

protected void RecuperarPass()
{
    if (TextBoxCorreoRecovery.Text.Length == 0)
    {
        TextBoxRespuestaRev.Text = "";
        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Information,
"Campo esta vacio");
    }
    else if (Int32.Parse(ddlPreguntaRecovery.SelectedValue) == 0)
    {
        TextBoxRespuestaRev.Text = "";
        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Information,
"Campo esta vacio");
    }
    else if (TextBoxRespuestaRev.Text.Length == 0)
    {
        TextBoxRespuestaRev.Text = "";
    }
}

```

```

        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Information,
"Campo esta vacio");
    }
    else
    {
        if (email_bien_escrito(TextBoxCorreoRecovery.Text))
        {
            string result;

            /* Se abre una Conexion con la base de datos*/

            Conexion conexion = new Conexion();
            using (SqlConnection con = Conexion.Conectar())
            {

                /* Nombre del procedimiento almacenado con sus parámetros*/

                using (SqlCommand cmd = new SqlCommand("Validacion_PassRecovery"))
                {
                    cmd.CommandType = CommandType.StoredProcedure;
                    cmd.Parameters.AddWithValue("@email", TextBoxCorreoRecovery.Text);
                    cmd.Parameters.AddWithValue("@idPregunta",
Int32.Parse(ddlPreguntaRecovery.SelectedValue));
                    cmd.Parameters.AddWithValue("@respuesta", TextBoxRespuestaRev.Text);
                    cmd.Connection = con;
                    con.Open();
                    result = Convert.ToString(cmd.ExecuteScalar());
                    con.Close();
                }

                switch (result)
                {
                    case "0":
                        // Response.Write("<script LANGUAGE='JavaScript' >alert('Debe ingresar los datos
correctos')</script>");
                        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Debe ingresar
los datos correctos");
                        break;

                    case "-1":
                        // Response.Write("<script LANGUAGE='JavaScript' >alert('Error al comparar
respuestas')</script>");
                        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Error al
comparar respuestas");
                        break;
                    case "-2":
                        // Response.Write("<script LANGUAGE='JavaScript' >alert('Cuenta no
activa')</script>");
                        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Error, "Cuenta no
activa");
                        break;
                    case "-3":
                        // Response.Write("<script LANGUAGE='JavaScript' >alert('Usuario no
existe')</script>");
                        MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Error, "Usuario no
existe");

```

break;

default:

```

        MailMessage Msg = new MailMessage();
        // Sender e-mail address.
        Msg.From = new MailAddress(TextBoxCorreoRecovery.Text);
        /* Diseño del Correo a enviar al usuario (Codigo HTML) */
        Msg.To.Add(TextBoxCorreoRecovery.Text);
        Msg.Subject = "Recuperacion de clave";
        Msg.Body =
            "<html>" +
            "<head>" +
            "<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\">" +
            "<title>Documento sin título</title>" +
            "<style type=\"text/css\">" +
            "body {" +
            "margin-left: 0px;" +
            "margin-top: 0px;" +
            "margin-right: 0px;" +
            "margin-bottom: 0px;" +
            "}" +
            "body,td,th {" +
            "color: #FFF;" +
            "font-family: Arial, Helvetica, sans-serif;" +
            "font-weight: bold;" +
            "font-size: 13px;" +
            "}" +
            "body,h2 {" +
            "color: #000;" +
            "font-family: Arial, Helvetica, sans-serif;" +
            "font-weight: bold;" +
            "font-size: 14px;" +
            "padding-left:10px;" +
            "}" +
            "body,h3 {" +
            "color: #06C;" +
            "font-family: Arial, Helvetica, sans-serif;" +
            "font-weight: bold;" +
            "font-size: 14px;" +
            "padding-left:10px;" +
            "}" +
            "body,h4 {" +
            "color: #666;" +
            "font-family: Arial, Helvetica, sans-serif;" +
            "font-weight: bold;" +
            "font-size: 13px;" +
            "padding-left:10px;" +
            "}" +
            "body,pietpagina{" +
            "color: #666;" +
            "font-family: Arial, Helvetica, sans-serif;" +
            "font-weight: bold;" +
            "font-size: 13px;" +
            "}" +
            "" +
            "" +

```

```

        "</style>" +
        "</head>" +
        "" +
        "<body>" +
        "<table width=\"50%\" border=\"0\" align=\"left\" cellpadding=\"3\"
cellspacing=\"0\" style=\"border:solid 1px; color:#06C; margin:5px; \">" +
        "<tr>" +
        "<td bgcolor=\"#0066CC\"><table width=\"100%\" border=\"0\"
cellspacing=\"5\" cellpadding=\"0\">" +
        "<tr>" +
        "<td width=\"88%\">Notificación de Correo Electrónico</td>" +
        "<td width=\"12%\" align=\"left\">&nbsp;</td>" +
        "</tr>" +
        "</table></td>" +
        "</tr>" +
        "<tr>" +
        "<td height=\"100px\" valign=\"top\">" +
        "<div><h3>Estimad@ </h3></div>" +
        "<div><h3>Porfavor, verifica los detalles de tu cuenta</h3></div>" +
        "<div><h3> La contraseña de su usuario <i></i> es: <h2>" +
        /*Decodifica la respuesta a lenguaje humano*/
Base64_Decode(result) + "</h2><br/><br/><b><u></u></b></p></h3></div>" +
        // "<div><h4>Gracias,</h4></p></div>"
        "</td>" +
        "</tr>" +
        "<tr>" +
        "<td align=\"center\" bgcolor=\"#0066CC\" ><div style=\" font-
size:10px;\">UNI derechos reservados</div></td>" +
        "</tr>" +
        "</table>" +
        "</body>" +
        "</html>");
    
```

```
Msg.IsBodyHtml = true;
```

```
/*Informacion del Correo Emisor */
```

```

        SmtplibClient smtp = new SmtplibClient();
        smtp.Host = "smtp.gmail.com";
        smtp.Port = 587;
        smtp.Credentials = new
    
```

```
/*Correo de la institución y contraseña para poder enviar el correo*/
```

```

        System.Net.NetworkCredential("xxxxxxxxxxxxxx", "xxxxxxxxxxxxxx");
        smtp.EnableSsl = true;
        try
        {
            smtp.Send(Msg);
            ScriptManager.RegisterStartupScript(this,
this.GetType(), "Aviso", " $( \"#dialog\" ).dialog();", true);
        }
        catch (System.Net.Mail.SmtpException ex)
        {

            Console.WriteLine(ex.Message);
            Console.ReadLine();
        }
    
```



```

        ScriptManager.RegisterStartupScript(this,
this.GetType(), "Error", " $( \".#dialog\" ).dialog();", true);
    }

    // Clear the textbox values
    TextBoxCorreoRecovery.Text = "";
    mvProductos.SetActiveView(vgrid);

    MessageBoxSafe.SetMessage(App_Controls_MessageBox.MessageType.Success, "Se ha
enviado un Correo a su cuenta!");
    break;
    }
}

else
{
    TextBoxCorreoRecovery.Text = "";
    mvProductos.SetActiveView(vgrid);
    MessageBoxRC.SetMessage(App_Controls_MessageBox.MessageType.Success,
"Ingrese un correo válido!");
}
}
}

```

## 25. CRUD PARA EL SISTEMA WEB

### 25.1 CRUD DE TRÁMITES ACADÉMICOS

La creación de las funcionalidades básicas para Crear, Leer y Actualizar. Se tomará como ejemplo el proceso de trámites académicos.

### 25.2 DISEÑO DEL FORMULARIO DE WFP\_documento.aspx

Para el diseño de los formularios se ocupó el web form designer para la integración de los componentes del formulario, estilos CSS 3, librerías de Bootstrap (Para el redimensionamiento de los formularios) y JQuery.

```

<div>
    <div class="x-panel-hd">
        
        <span>Listado de Documentos</span>
    </div>

```

```

</div>
<br />
<asp:MultiView ID="myDoc" runat="server">
  <asp:View ID="vgrid" runat="server">
    <script type="text/javascript">
      $(document).ready(function () {
        var table = $('#example').DataTable();
        // Setup - add a text input to each footer cell
        $('#example tfoot th').each(function () {
          var title = $('#example thead
th').eq($(this).index()).text();
          $(this).html('<input type="text" placeholder="busqueda '
+ title + '" />');
        });

        // DataTable

        $('#example tbody').on('click', 'tr', function () {

          var idx = table.row(this).index();
          // alert(idx);
          $('#MainContent_hfPosition').val(idx);
          if ($(this).hasClass('selected')) {
            $(this).removeClass('selected');
          }
          else {
            table.$('tr.selected').removeClass('selected');
            $(this).addClass('selected');
          }
        });

        // Apply the search
        table.columns().eq(0).each(function (colIdx) {
          $('#input', table.column(colIdx).footer()).on('keyup
change', function () {
            table
            .column(colIdx)
            .search(this.value)
            .draw();
          });
        });
      });
    </script>
    <!-- <asp:ToolkitScriptManager ID="ToolkitScriptManager1"
runat="server">
      </asp:ToolkitScriptManager>--%>
    <!--<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>--%>
    <asp:ModalPopupExtender ID="ModalPopupExtender1" runat="server"
PopupControlID="Pan11"
      TargetControlID="btnAssignReque" CancelControlID="Button2"
BackgroundCssClass="Background">
      </asp:ModalPopupExtender>
    <asp:Panel ID="Pan11" runat="server" CssClass="Popup" align="center"
Style="display: none">
      <iframe style="width: 700px; height: 300px;" id="irm1"
src="DetailRequisitos.aspx"
        runat="server"></iframe>
    <br />
  </asp:View>
</asp:MultiView>

```

```

        <asp:Button ID="Button2" runat="server" Text="Cerrar" />
    </asp:Panel>

    <div class="toolbar">
        <asp:Button ID="btnAddDoc" CssClass="btn_nuevo" runat="server"
Text="Nuevo" OnClick="btnAdd_Click" />
        <asp:Button ID="btnEditarDoc" runat="server"
CssClass="btn_editar" OnClick="btnEditar_Click"
Text="Editar" />
        <asp:Button ID="btnAssignReque" runat="server" Text="Asignar
Requisito" CssClass="btn_assign"
Width="151px" />
        <asp:HiddenField ID="hfPosition" runat="server" />
    </div>
    <div>
        <uc1:MessageBox ID="MessageBoxSave" runat="server" />
    </div>
    <div>
        <uc1:MessageBox ID="MessageBoxUpdate" runat="server" />
    </div>
    <div class="well">
        <asp:Repeater ID="rptDoc" runat="server">
            <HeaderTemplate>
                <table id="example" class="table table-striped table-
bordered" cellpadding="0">
                    <thead>
                        <tr>
                            <th >
                                Nombre del documento
                            </th>
                            <th >
                                Tipo de documento
                            </th>
                            <th >
                                Precio
                            </th>
                            <th >
                                Tiempo
                            </th>
                            <th >
                                Activo
                            </th>
                        </tr>
                    </thead>
                    <tfoot>
                        <th style="display: none;">
                            Nombre
                        </th>
                        <th style="display: none;">
                            Tipo de documento
                        </th>
                        <th style="display: none;">
                            Precio
                        </th>
                        <th style="display: none;">

```

```

        Tiempo
    </th>
    <th style="display: none;">
        Activo
    </th>
</tr>
</tfoot>
<tbody>
</HeaderTemplate>
<ItemTemplate>
    <tr>
        <td>
            <%#Eval("nombre_documento") %>
        </td>
        <td>
            <%#Eval("Catalogo.nombre") %>
        </td>
        <td>
            <%# Eval("precio") %>
        </td>
        <td>
            <%# Eval("Catalogo1.nombre") %>
            <asp:Image ID="Image1" runat="server"
ImageUrl='<%# (Eval("activo").Equals(true) ? "~/img/active.png" :
(Eval("activo").Equals(false) ? "~/img/inactive.png" : "~/img/eleminated.png"))%>'
/>
        </td>
    </tr>
</ItemTemplate>
<FooterTemplate>
</tbody> </table>
</FooterTemplate>
</asp:Repeater>
</div>

</asp:View>
<asp:View ID="vAdd" runat="server">
    <div class="well">
        <table class="table" align="center">
            <tr>
                <td >
                    Nombre:
                </td>
                <td >
                    <asp:TextBox ID="txtNombreDoc" runat="server"
Height="32px" Width="200px"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <td >
                    Descripcion:
                </td>
                <td >
                    <asp:TextBox ID="txtDescripcionDoc" runat="server"
Height="80px" Width="200px" TextMode="MultiLine"></asp:TextBox>
                </td>
            </tr>
        </table>
    </div>
</asp:View>

```

```

        </tr>
        <tr>
            <td >
                Precio:
            </td>
            <td>
                <asp:TextBox ID="textBoxPrecioDoc" runat="server"
Height="32px" Width="200px"></asp:TextBox>
                <asp:RegularExpressionValidator
ID="RegularExpressionValidator1" runat="server" ControlToValidate="textBoxPrecioDoc"
                ErrorMessage="Ingresa solo Numeros"
ValidationExpression="^\d+$"
ValidationGroup="check"></asp:RegularExpressionValidator>
            </td>
        </tr>
        <tr>
            <td >
                Tiempo del tramite:
            </td>
            <td >
                <asp:DropDownList ID="ddlTiempo" runat="server"
DataTextField="nombre"
                DataValueField="id">
            </asp:DropDownList>
            </td>
        </tr>
        <tr>
            <td >
                Activo:
            </td>
            <td >
                <asp:CheckBox ID="cbActivoDoc" runat="server"
Text="Activo" />
            </td>
        </tr>
        <tr>
            <td >
                Tipo Documento:</td>
            <td >
                <asp:DropDownList ID="ddlTipodoc" runat="server"
DataTextField="nombre"
                DataValueField="id">
            </asp:DropDownList>
            </td>
        </tr>
    </table>
    <div class="toolbar">
        <asp:Button ID="btnRegresar" runat="server"
CssClass="btn_retun" OnClick="btnRegresar_Click"
                Text="Regresar" />
        <asp:Button ID="btnGuardar" runat="server"
CssClass="btn_save" OnClick="btnGuardar_Click"
                Text="Guardar" />
        <asp:Button ID="btnUpdate" runat="server"
CssClass="btn_update" OnClick="btnUpdate_Click"
                Text="Actualizar" />
    </div>

```

```

        <br />
        <div>
            <uc1:MessageBox ID="MessageBoxNotification" runat="server"
/>
        </div>
        <asp:HiddenField ID="hfId" runat="server" />
    </div>
</asp:View>
</asp:MultiView>
</div>
    <asp:LinkButton ID="lnkFake" runat="server" />
    <asp:ModalPopupExtender ID="mpeTimeout" BehaviorID="mpeTimeout"
runat="server" PopupControlID="pnlPopup"
    TargetControlID="lnkFake" OkControlID="btnYes" CancelControlID="btnNo"
    BackgroundCssClass="modalBackground"
    OnOkScript="ResetSession()">
</asp:ModalPopupExtender>
    <asp:Panel ID="pnlPopup" runat="server" CssClass="modalPopup" Style="display:
none">
        <div class="header">
            Sesión Expirando!
        </div>
        <div class="body">
            Su sesión expirará en &nbsp;<span id="seconds"></span>&nbsp;segundos.<br
/>
            Quieres reiniciar el tiempo de sesión?
        </div>
        <div class="footer" align="right">
            <asp:Button ID="btnYes" runat="server" Text="Si" CssClass="yes" />
            <asp:Button ID="btnNo" runat="server" Text="No" CssClass="no" />
        </div>
    </asp:Panel>

```

## 25.3 INSERCIÓN

Este método realiza la inserción de la nueva información en la base de datos.

Primeramente se activa la vista donde se muestran los campos vacíos donde se escribirá la nueva información necesaria para almacenarla. (Fig. 1)

```

protected void btnAdd_Click(object sender, EventArgs e)
{
    mvDoc.SetActiveView(vAdd);
    cbActivoDoc.Checked = false;
    txtNombreDoc.Text = "";
    txtDescripcionDoc.Text = "";
    textBoxPrecioDoc.Text = "";
    ddlTiempo.SelectedIndex = 0;
    ddlTipodoc.SelectedIndex = 0;
    btnUpdate.Enabled = false;
    btnGuardar.Enabled = true;
}

```

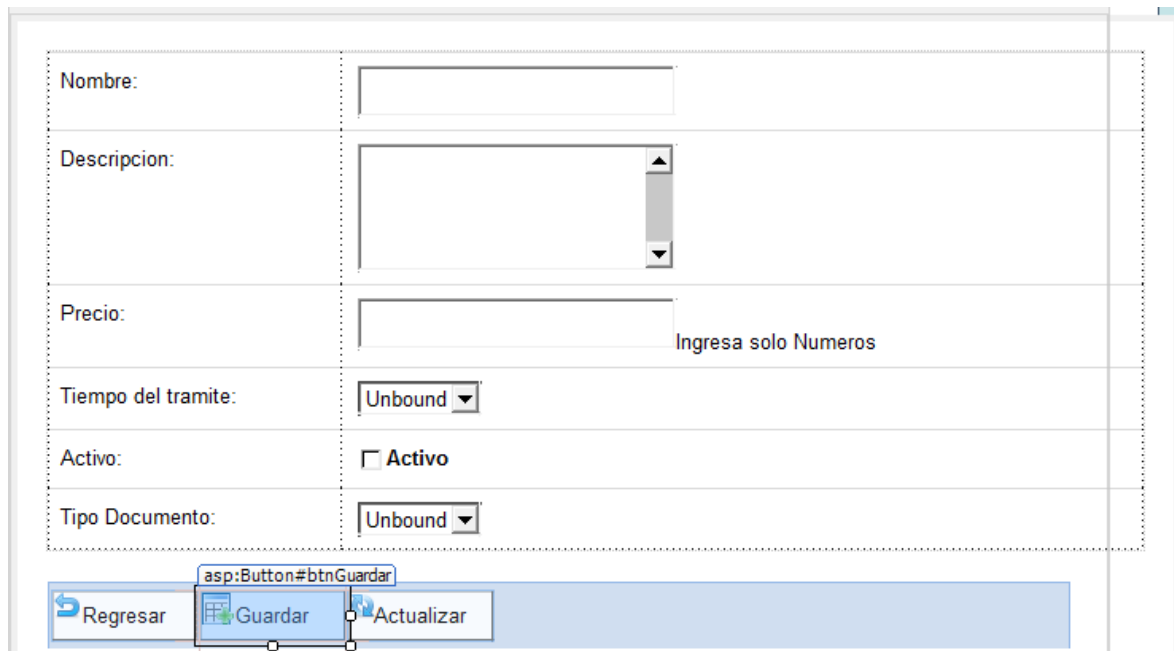


Fig. 1. Activación de la vista de almacenamiento

### 25.3.1 FUNCION DE INSERCIÓN DE NUEVOS DATOS.

Esta función realiza la validación de los campos que no estén vacíos y que contengan el contenido requerido, luego de la validación se insertan los datos mediante tecnología LINQ tras el evento del botón Guardar.

```
protected void btnGuardar_Click(object sender, EventArgs e)
{
    if (txtNombreDoc.Text != string.Empty && textBoxPrecioDoc.Text !=
string.Empty &&
        Int32.Parse(ddlTiempo.SelectedValue) != 0 &&
Int32.Parse(ddlTipodoc.SelectedValue) != 0)
    {
        c_documento.insertarDoc(txtNombreDoc.Text,
            txtDescripcionDoc.Text,
            Double.Parse(textBoxPrecioDoc.Text),
            Int32.Parse(ddlTiempo.SelectedValue),
            cbActivoDoc.Checked, Int32.Parse(ddlTipodoc.SelectedValue));
        loadGridDocs();
        CargarCatalogo();
        mvDoc.SetActiveView(vgrid);
        MessageBoxSave.SetMessage(App_Controls_MessageBox.MessageType.Success,
"Registro almacenado");
    }
}
```

```

        else
        {
            MessageBoxNotification.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Debe
            de llenar los campos solicitados");
            mvDoc.SetActiveView(vAdd);
        }
    }
}

```

### 25.3.2 FUNCION DE INSERCIÓN LINQ

Usando la tecnología del Framework LINQ la inserción de datos se realiza utilizando funciones propias para la inserción, edición y visualización de datos.

```

public int insertarDoc(string nombre, string descripcion, double precio, int
tiempo, bool activo, int tipo_doc)
{
    Documento documento = new Documento();
    documento.nombre_documento = nombre;
    documento.descripcion = descripcion;
    documento.precio = precio;
    documento.tiempo = tiempo;
    documento.activo = activo;
    documento.tipo_documento = tipo_doc;
    /*METODO PARA INSERTAR*/
    contexto.Documentos.InsertOnSubmit(documento);
    contexto.SubmitChanges();
    return 0;
}

```

### 25.4 EDICIÓN

Este método realiza la edición de información almacenada en la base de datos.

Primeramente se activa la vista donde se muestran los campos vacíos donde se presentará la nueva información necesaria para ser editada. (Fig. 2)



### 25.4.1 FUNCION DE EDICION DE DATOS.

Esta función realiza la validación de los campos contengan la información sobre el registro que será modificado, luego de la edición del contenido se actualizará los datos mediante tecnología LINQ tras el evento del botón Actualizar.

```
protected void btnUpdate_Click(object sender, EventArgs e)
{
    if (txtNombreDoc.Text != string.Empty && textBoxPrecioDoc.Text !=
string.Empty &&
        Int32.Parse(ddlTiempo.SelectedValue) != 0 &&
Int32.Parse(ddlTipodoc.SelectedValue) != 0)
    {
        Documento documento = new Documento();
        documento.id = int.Parse(hfId.Value);
        documento.nombre_documento = txtNombreDoc.Text;
        documento.tiempo = Int32.Parse(ddlTiempo.SelectedValue);
        documento.descripcion = txtDescripcionDoc.Text;
        documento.precio = Double.Parse(textBoxPrecioDoc.Text);
        documento.activo = cbActivoDoc.Checked;
        documento.tipo_documento = Int32.Parse(ddlTipodoc.SelectedValue);

        /*METODO PARA ACTUALIZAR*/
        c_documento.actualizarDocumento(documento);
        loadGridDocs();
        mvDoc.SetActiveView(vgrid);

        MessageBoxUpdate.SetMessage(App_Controls_MessageBox.MessageType.Information,
"Registro Modificado");
    }
    else
    {
        MessageBoxNotification.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Debe
de llenar los campos solicitados");
        mvDoc.SetActiveView(vAdd);
    }
}
```

### 25.4.2 FUNCION DE INSERCIÓN LINQ

Usando la tecnología del Framework LINQ la inserción de datos se realiza utilizando funciones propias para la inserción, edición y visualización de datos.

```
public int actualizarDocumento(Documento A)
{
    //Get Single course which need to update
    Documento documento = contexto.Documentos.Single(mp => mp.id == A.id);
    documento.nombre_documento = A.nombre_documento;
    documento.descripcion = A.descripcion;
    documento.precio = A.precio;
    documento.tiempo = A.tiempo;
    documento.activo = A.activo;
    documento.tipo_documento = A.tipo_documento;
    contexto.SubmitChanges();
    return 1;
}
```

### 25.4.3 VISUALIZACION DE DATOS

El proceso de visualización muestra los datos registrados en la base de datos en dependencia de lo que se requiere.

### 25.4.4 FUNCION DE VISUALIZACION

Esta función muestra en el Grid la información más importante de acuerdo a cada requerimiento. Esta información se carga automáticamente al abrir cada formulario.

```
private void loadGridDocs()
{
    rptDoc.DataSource = c_documento.ListarDocDw();
    rptDoc.DataBind();
}
```

## 26. FUNCION DE VISUALIZACION LINQ

Esta función muestra todo el contenido requerido de datos mediante tecnología LINQ tras haberse cargado la página.

```
public List<Documento> ListarDocDw()
{
    return (from p in contexto.Documentos
            where p.activo.Equals(true)
            select p).ToList();
}
```

## 27. PROCESAMIENTO DE TRAMITES ACADEMICOS

Este proceso se realiza para cambiar de estado los trámites en la bandeja de entrada, de recibido a en proceso y luego a concluidos, este proceso lo realizan solamente el personal con privilegios.

### 27.1 FUNCION DE VERIFICACION DE PERFIL

Este proceso verifica si el perfil que está registrado en el instante tiene los privilegios para procesar un trámite académico localizado en la bandeja de entrada de solicitudes.

```
private void Perfil_Procesa()
{
    /*Se abre una nueva conexión con la base de datos */
    Conexion conexion = new Conexion();
    using (SqlConnection con = Conexion.Conectar())
    {
        /*Nombre del procedimiento almacenado y los parámetros */
        using (SqlCommand cmd = new SqlCommand("Perfil_Procesa"))
        {
            texto = Session["Perfil"].ToString();
            cmd.CommandType = CommandType.StoredProcedure;
        }
    }
}
```

```

        cmd.Parameters.AddWithValue("@Id_perfil",
Session["Perfil"].ToString());
        cmd.Connection = con;
        con.Open();
        Procesa = Convert.ToBoolean(cmd.ExecuteScalar());
        con.Close();
    }
}
}

```

## 27.2 FUNCION PARA PROCESAR TRÁMITES

Este método cambia de estado los trámites académicos en dependencia de los privilegios y perfil que tenga asignado el usuario. Este método se lleva acabo tras el evento del botón Process, verifica el perfil y luego verifica el límite del privilegio.

```

protected void btnProcess_Click(object sender, EventArgs e)
{
    if (hfPositionProcess.Value == "")
    {
        Response.Write("<script LANGUAGE='JavaScript' >alert('Debe seleccionar el registro a modificar')</script>");
        return;
    }
    else
    {
        int estado = 0;

        Tramite tramite =
c_tramite.obtenerTramite(int.Parse(hfPositionProcess.Value),
Convert.ToInt32(Session["Perfil"]));

        hfIdProcess.Value = tramite.id.ToString();
        tramite.nombre = tramite.nombre;
        tramite.fecha_inicio = tramite.fecha_inicio;
        tramite.fecha_fin = tramite.fecha_fin;
        estado = int.Parse(tramite.estado.ToString());

        Perfil_Procesa();

        if (Procesa)
        {

            if (estado == "No tramitado")
            {

```

```

        tramite.estado = "En proceso";
        tramite.activo = tramite.activo;
        c_tramite.actualizarTramite(tramite);
        hfPositionProcess.Value = null;
        //Response.Write("<script LANGUAGE='JavaScript' >alert('Tramite
Procesado')</script>");

MessageBoxSucces.SetMessage(App_Controls_MessageBox.MessageType.Success, "Trámite
Procesado");
        loadTramiteList();
    }

    else if (estado == "En proceso");
    {

        tramite.estado = "Procesado";

        tramite.activo = tramite.activo;
        c_tramite.actualizarTramite(tramite);
        hfPositionProcess.Value = null;

        BindInfoTramite2(Int32.Parse(hfIdProcess.Value));
        EnviarNotifcation();
        loadTramiteList();
        //Response.Write("<script LANGUAGE='JavaScript' >alert('Tramite
Procesado')</script>");

        MessageBoxSucces.SetMessage(App_Controls_MessageBox.MessageType.Success, "Trámite
Procesado");

    }
    else
    {

        MessageBoxInfo.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Trámite ya
esta en bandeja del encargado");
    }
}

    else if (estado == "No tramitado"))
    {
        tramite.estado = "En proceso";
        tramite.activo = tramite.activo;
        c_tramite.actualizarTramite(tramite);
        hfPositionProcess.Value = null;

        MessageBoxSucces.SetMessage(App_Controls_MessageBox.MessageType.Success, "Trámite
Procesado");
        loadTramiteList();
    }
}

```

```

        else
        {
            MessageBoxInfo.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Trámite ya
            esta en bandeja del encargado");
        }
    }
}

```

## 28. REALIZAR UN NUEVO TRAMITE

El proceso de solicitud de trámites académicos lo realiza únicamente el usuario que se haya registrado exitosamente en el sistema y que tenga su cuenta activa para poder realizar trámites desde el sistema.

### 28.1 WIZARD

Para este proceso se utilizó un componente de asp.net llamado WIZARD con el objetivo de promover el orden y cuya presentación de la información sea intuitiva para el usuario. Este es desarrollado en asp.net y JQuery. (Fig. 3 y 4)

```

<script type="text/javascript">
    $(document).ready(function () {

        $('#wizard').smartWizard();

        function onFinishCallback() {
            $('#wizard').smartWizard('showMessage', 'Finish Clicked');
            alert('Finish Clicked');
        }
    });

</script>

```

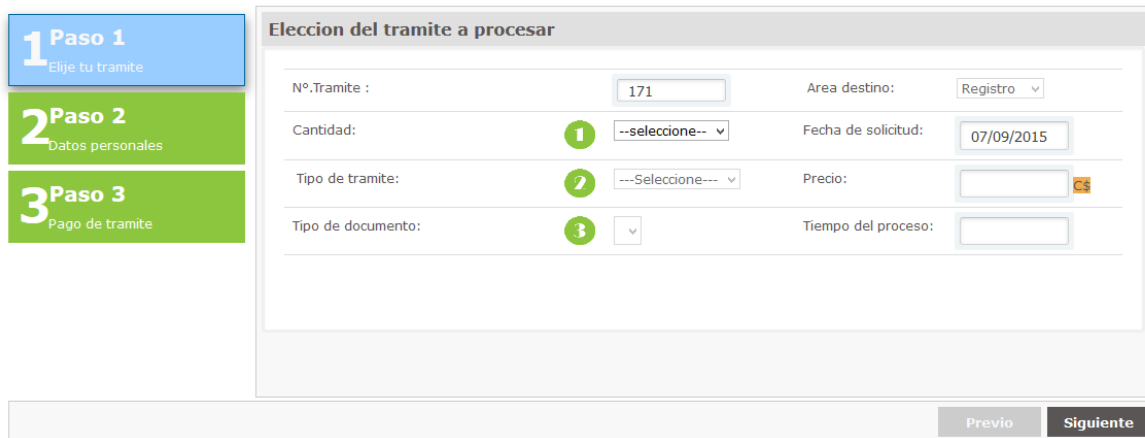


Fig. 3 Wizard Etapa 1



Fig. 4 Wizard Etapa 3

## 28.2 CONSTRUCCION DEL WIZARD

Este es el código asp.net que muestra manejo del Wizard en el ambiente de desarrollo.

```
<asp:MultiView ID="MPago" runat="server">
  <asp:View ID="VFill" runat="server">
    <div>
      <uc1:MessageBox ID="MessageBoxNotification" runat="server" />
    </div>
    <div id="wizard" class="swMain">
      <ul>
        <li><a href="#step-1">
          <label class="stepNumber">
            1</label>
          <span class="stepDesc">Paso 1<br />
            <small>Elige tu tramite</small></span></a></li>
        <li><a href="#step-2">
          <label class="stepNumber">
```

```

                2</label>
                <span class="stepDesc">Paso 2<br />
                <small>Datos personales</small></span></a></li>
            <li><a href="#step-3">
                <label class="stepNumber">
                    3</label>
                <span class="stepDesc">Paso 3<br />
                <small>Pago de tramite</small> </span></a></li>
        </ul>
        <div id="step-1">
            <h2 class="StepTitle">
                Eleccion del tramite a procesar</h2>
            <asp:UpdatePanel ID="UpdatePanel1" runat="server">
                <ContentTemplate>
                    <div class="well">
                        <table class="table" align="center">
                            <tr>
                                <td class="style6" colspan="2">
                                    N°.Tramite :
                                </td>
                                <td>
                                    <asp:TextBox ID="TextBoxNum"
runat="server" Width="106px" Enabled="False"></asp:TextBox>
                                </td>
                                <td>
                                    &nbsp;   Area destino:
                                </td>
                                <td>
                                    <asp:DropDownList ID="ddlArea"
runat="server" DataTextField="nombre_area" DataValueField="id_area"
                                    Enabled="False">
                                        </asp:DropDownList>
                                </td>
                            </tr>
                            <tr>
                                <td class="style3">
                                    Cantidad:
                                </td>
                                <td>
                                    <asp:Image ID="Image5" runat="server"
ImageUrl="~/img/images/Step1.png" />
                                </td>
                                <td class="style5">
                                    <asp:DropDownList ID="ddlCantidad"
runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ddlCantidad_SelectedIndexChanged">
                                        <asp:ListItem Value="0">--
seleccione--</asp:ListItem>
                                        <asp:ListItem>1</asp:ListItem>
                                        <asp:ListItem>2</asp:ListItem>
                                        <asp:ListItem>3</asp:ListItem>
                                        <asp:ListItem>4</asp:ListItem>
                                        <asp:ListItem>5</asp:ListItem>
                                        <asp:ListItem>6</asp:ListItem>
                                        <asp:ListItem>7</asp:ListItem>
                                        <asp:ListItem>8</asp:ListItem>
                                        <asp:ListItem>9</asp:ListItem>
                                        <asp:ListItem>10</asp:ListItem>

```



```

        </asp:DropDownList>
    </td>
    <td class="style7">
        Fecha de solicitud:
    </td>
    <td class="style6">
        <asp:TextBox ID="TextBoxFechaSoli"
runat="server" Enabled="False" Width="106px" Height="28px"></asp:TextBox>
    </td>
</tr>
<tr>
    <td class="style3">
        &nbsp;  Tipo de tramite:
    </td>
    <td>
        <asp:Image ID="Image6" runat="server"
ImageUrl="~/img/images/Step2.png" />
    </td>
    <td class="style5">
        <asp:DropDownList ID="ddlTipotramite"
runat="server" DataTextField="nombre" DataValueField="id"
AppendDataBoundItems="True"
Enabled="False" AutoPostBack="True"
OnSelectedIndexChanged="ddlTipotramite_SelectedIndexChanged"
EnableTheming="True">
        </asp:DropDownList>
    </td>
    <td>
        Precio:
    </td>
    <td>
        <asp:TextBox ID="TextBoxPrecio"
runat="server" AutoPostBack="True" Enabled="False"
Height="27px"
Width="106px"></asp:TextBox>
        <asp:Label ID="Label3" runat="server"
CssClass="label-warning" Text="C$"></asp:Label>
    </td>
</tr>
<tr>
    <td class="style3">
        Tipo de documento:
    </td>
    <td>
        <asp:Image ID="Image7" runat="server"
ImageUrl="~/img/images/Step3.png" />
    </td>
    <td class="style5">
        <asp:DropDownList ID="ddlTipoDoc"
runat="server" DataTextField="nombre_documento"
DataValueField="id"
OnSelectedIndexChanged="ddlTipoDoc_SelectedIndexChanged"
AutoPostBack="True"
AppendDataBoundItems="True" Enabled="False" Height="28px">
        </asp:DropDownList>
    </td>
    <td>
        Tiempo del proceso:
    </td>
    <td>
        <asp:Image ID="Image8" runat="server"
ImageUrl="~/img/images/Step4.png" />
    </td>
</tr>
</table>

```

[illegible]

```

        Facultad:
    </td>
    <td>
        <asp:TextBox ID="TextBoxFacultad" runat="server"
Height="25px" Width="199px" Enabled="False"></asp:TextBox>
    </td>
</tr>
<tr>
    <td class="style3">
        Carrera:
    </td>
    <td class="style5">
        <asp:TextBox ID="TextBoxCarrera" runat="server"
Height="24px" Width="199px" Enabled="False"></asp:TextBox>
    </td>
    <td>
        &nbsp;
    </td>
    <td>
        &nbsp;
        <asp:TextBox ID="TextTemp2" runat="server"
Visible="False"></asp:TextBox>
        <br />
        <asp:TextBox ID="TextBoxTemp" runat="server"
Visible="False"></asp:TextBox>
    </td>
</tr>
</table>
</div>
</div>
<div id="step-3">
    <div id="tabs">
        <ul>
            <li><a href="#tabs-1">Pago con Tarjeta de
debito</a></li>
        </ul>
        <div id="tabs-1">
            <div>
                <asp:Label ID="Label8" runat="server" Text="
Ingresa los datos de tu tarjeta para realizar el pago."
Font-Size="Small"></asp:Label>
            </div>
            <br />
            <div style="background-color: #31AECE;">
                <asp:Label ID="Label14" runat="server" Text="Tarjeta
de debito / credito" Font-Bold="True"
ForeColor="White"></asp:Label>
            </div>
            <asp:Label ID="Label15" runat="server" Text="(*) Emisor
de la Tarjeta : " Font-Size="Small"
Font-Bold="False"></asp:Label>
            <asp:DropDownList ID="ddlTarjeta" runat="server"
DataTextField="nombre" DataValueField="id">
            </asp:DropDownList>
            <asp:Image ID="Image2" runat="server" Height="27px"
ImageUrl="~/img/american.gif"
Width="39px" />

```

```

        <asp:Image ID="Image3" runat="server" Height="27px"
ImageUrl=~\img\visa.gif" Width="37px" />
        <asp:Image ID="Image4" runat="server" Height="27px"
ImageUrl=~\img\mastercard.gif"
            Width="45px" />
        <asp:RequiredFieldValidator ID="RequiredFieldValidator1"
runat="server" ControlToValidate="ddlTarjeta"
            ErrorMessage="Debes elegir una opcion!"
InitialValue="---Seleccione---">
        </asp:RequiredFieldValidator>
        <br />
        <br />
        <asp:Label ID="Label6" runat="server" Text="(*) Numero
de la Tarjeta : " Font-Size="Small"
            Font-Bold="False"></asp:Label>
        <asp:TextBox ID="TextBoxPart1" runat="server"
Width="65px" MaxLength="4" Height="24px"></asp:TextBox>-<asp:TextBox
            ID="TextBoxPart2" runat="server" Width="65px"
MaxLength="4" Height="24px"></asp:TextBox>-
        <asp:TextBox ID="TextBoxPart3" runat="server"
Width="65px" MaxLength="4" Height="24px"></asp:TextBox>-<asp:TextBox
            ID="TextBoxPart4" runat="server" Width="65px"
MaxLength="4" Height="24px"></asp:TextBox><br />
        <br />
        <asp:Label ID="Label9" runat="server" Text="(*) Fecha de
Caducidad : " Font-Size="Small"
            Font-Bold="False"></asp:Label>
        <asp:DropDownList ID="ddlyear" runat="server">
        </asp:DropDownList>
        <asp:DropDownList ID="ddlmes" runat="server">
            <asp:ListItem Value="0">Mes</asp:ListItem>
            <asp:ListItem>1</asp:ListItem>
            <asp:ListItem>2</asp:ListItem>
            <asp:ListItem>3</asp:ListItem>
            <asp:ListItem>4</asp:ListItem>
            <asp:ListItem>5</asp:ListItem>
            <asp:ListItem>6</asp:ListItem>
            <asp:ListItem>7</asp:ListItem>
            <asp:ListItem>8</asp:ListItem>
            <asp:ListItem>9</asp:ListItem>
            <asp:ListItem>10</asp:ListItem>
            <asp:ListItem>11</asp:ListItem>
            <asp:ListItem>12</asp:ListItem>
        </asp:DropDownList>
        <asp:RequiredFieldValidator ID="RequiredFieldValidator3"
runat="server" ControlToValidate="ddlmes"
            ErrorMessage="Debes elegir un mes!"
InitialValue="Mes">
        </asp:RequiredFieldValidator>
        <asp:RequiredFieldValidator ID="RequiredFieldValidator2"
runat="server" ControlToValidate="ddlyear"
            ErrorMessage="Debes elegir un año!"
InitialValue="Año">
        </asp:RequiredFieldValidator>
        <br />
        <br />
        <asp:Label ID="Label10" runat="server" Font-Bold="False"
Font-Size="Small" Text="(*) Codigo de seguridad (CVV) :"></asp:Label>

```

```

        <asp:TextBox ID="TextBoxCVV" runat="server"
Height="24px" MaxLength="3" Width="65px"></asp:TextBox>
        <asp:RequiredFieldValidator ID="RequiredFieldValidator4"
runat="server" ControlToValidate="TextBoxCVV"
        ErrorMessage="Digite su numero de seguridad"
InitialValue="">
        </asp:RequiredFieldValidator>
        <br />
        <br />
        <div align="center">
            <asp:Button ID="ButtonGenerar" runat="server"
OnClick="ButtonGenerar_Click" Text="Generar Detalle"
            type="submit" />
        </div>
    </div>
</div>
</div>
</div>
</asp:View>
<asp:View ID="VDetallePago" runat="server">
    <br />
    <div style="text-align: center;">
        <asp:Button ID="btnImprimir" runat="server" OnClientClick="return
PrintPanel();"
            Text="Imprimir" />
    </div>
    <br />
    <div style="background-color: #CDE5EF; font-family: Arial, Helvetica,
sans-serif;
        font-weight: bold; color: #527881; text-align: center">
        Confirmación
    </div>
    <asp:Panel ID="pnlPopupDetalle" runat="server" Style="text-align:
center;">
        <br />
        <div style="border-style: solid; width: 50%; margin: 0 auto; text-
align: center;">
            <div style="background-color: #003366; style="text-align:
center;">
                <asp:Label ID="Label1" runat="server" Text="Detalle de
Transacción" Font-Bold="True"
                    ForeColor="White"></asp:Label>
            </div>
            <br />
            <table title="Detalle de Pago" style="width: 100%;"
class="styleTable">
                <tr>
                    <td>
                        &nbsp;
                    </td>
                    <td class="perfiles" colspan="3">
                        <table border="0" cellpadding="0" cellspacing="0"
width="100%" class="sorting_asc">
                            <tr>
                                <td class="modulo" align="center"
style="color: #000000; font-style: normal; font-weight: bold;
                                    font-size: large;">
                                    </td>

```

```

        </tr>
        <tr>
            <td class="modulo" align="center"
style="color: #000000; font-style: normal; font-weight: bold;
font-size: large;">
                Certificado de Pago
            </td>
        </tr>
    </table>
    <hr />
</td>
<td width="8">
    &nbsp;
</td>
</tr>
<tr>
<td>
    &nbsp;
</td>
<td valign="top" colspan="3">
    <table width="100%" style="border-style: groove">
        <tr>
            <td class="style11">
                Tramitante:
            </td>
            <td style="font-weight: bold">
                <asp:Label ID="LabelUser" runat="server"
Text="Label"></asp:Label>
            </td>
        </tr>
        <tr>
            <td class="style11">
                Carnet:
            </td>
            <td style="font-weight: bold">
                <asp:Label ID="LabelCarnet"
runat="server" Text="Label"></asp:Label>
            </td>
        </tr>
        <tr>
            <td class="modulo" colspan="2">
                &nbsp;
            </td>
            <tr>
                <td class="style11">
                    <asp:Label ID="Label27"
runat="server" CssClass="formu" Text="Concepto:"></asp:Label>
                </td>
                <td style="font-weight: bold">
                    <asp:Label ID="LbDocumento"
runat="server" CssClass="formu-btn2"></asp:Label>
                </td>
            </tr>
            <tr>
                <td class="style11">
                    <asp:Label ID="LbPrecio"
runat="server" CssClass="formu" Text="Precio:"></asp:Label>
                </td>

```

```

                                <td style="font-weight: bold">
                                    <asp:Label ID="Label131"
runat="server" Text="C$"></asp:Label>
                                <asp:Label ID="LbPrecioFinal"
runat="server" CssClass="formu-btn2"></asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td class="style11">
                                    Cantidad:
                                </td>
                                <td style="font-weight: bold">
                                    <asp:Label ID="LabelCantidad"
runat="server" Text="Label"></asp:Label>
                                </td>
                            </tr>
                            <tr>
                                <td class="modulo" colspan="2">
                                    <hr />
                                </td>
                            </tr>
                            <tr>
                                <td colspan="2" style="text-align:
center">
                                    <div>
                                        <table width="100%">
                                            <tr>
                                                <td class="style13">
                                                    Fecha de Solicitud :
                                                </td>
                                                <td style="font-weight:
bold" class="text-center">
                                                    <asp:Label
ID="lbFechaI" runat="server" CssClass="formu"></asp:Label>
                                                </td>
                                            </tr>
                                            <tr>
                                                <td class="style13">
                                                    Fecha de Entrega :
                                                </td>
                                                <td style="font-weight:
bold" class="text-center">
                                                    <asp:Label
ID="lbFechaEntrega" runat="server"></asp:Label>
                                                </td>
                                            </tr>
                                        </table>
                                    </div>
                                </td>
                            </tr>
                            <tr>
                                <td align="right" colspan="2"
valign="top">
                                    <hr />
                                </td>
                            </tr>
                        </tr>
                    </table>

```

[illegible]



```

        &nbsp;
    </td>
    <td bgcolor="#003060">
        &nbsp;
    </td>
</tr>
</table>
</div>
</asp:Panel>
<div class="footer" align="right">
    <asp:Button ID="btnYes" runat="server" Text="Si"
OnClick="btnYes_Click" OnClientClick="Confirm()" />
    <asp:Button ID="btnNo" runat="server" Text="No"
OnClick="btnNo_Click" />
</div>
<asp:ConfirmButtonExtender ID="Cbe" runat="server"
DisplayModalPopupID="Mpe" TargetControlID="btnYes">
</asp:ConfirmButtonExtender>

<asp:ModalPopupExtender ID="Mpe" runat="server"
PopupControlID="pnlPopup" TargetControlID="btnYes"
OkControlID="Button1" CancelControlID="Button2"
BackgroundCssClass="modalBackgroundE2">
</asp:ModalPopupExtender>
<asp:Panel ID="pnlPopup" runat="server" CssClass="modalPopupE2"
Style="display: none">
    <div class="headerE2">
        Verificacion
    </div>
    <div class="bodyE2">
        ¿ Desea realizar este tramite ?
    </div>
    <div class="footerE2" align="center">
        <asp:Button ID="Button1" runat="server" Text="Si" />
        <asp:Button ID="Button2" runat="server" Text="No" />
    </div>
</asp:Panel>

</asp:View>
</asp:MultiView>

```

### 28.3 CONCLUSIÓN DE TRÁMITE

Para el almacenamiento de la información del nuevo trámite se utiliza la esta función que verifica que cada uno de los campos requeridos no estén vacíos, valida información importante como número de tarjeta. Todo este proceso para generar el previo detalle de la transacción que se realizara, esto se realiza haciendo el llamado a un (Procedimiento almacenado).

```

protected void ButtonGenerar_Click(object sender, EventArgs e)
{
    /*Validacion de espacios vacios */

```

```

        if (TextBoxPart1.Text != string.Empty && TextBoxPart2.Text != string.Empty
&&
            TextBoxPart3.Text != string.Empty && TextBoxPart4.Text != string.Empty
&&
            TextBoxCVV.Text != string.Empty && TextBoxTemp.Text != string.Empty &&
TextTemp2.Text != string.Empty
            && Int32.Parse(ddlTipotramite.SelectedValue) != 0 &&
Int32.Parse(ddlTipoDoc.SelectedValue) != 0)
        {

/*Generación del Detalle de la transacción */

            total = (Int32.Parse(TextBoxTemp.Text) * float.Parse(TextTemp2.Text));
            LbMonto.Text = total.ToString();
            LbDocumento.Text = (ddlTipoDoc.SelectedItem.ToString());
            LbPrecioFinal.Text = float.Parse(TextTemp2.Text).ToString();
            lbFechaI.Text = DateTime.Now.ToShortDateString();
            DateTime newDate = DateTime.Now;
            lbFechaEntrega.Text = newDate.AddDays(2).ToShortDateString();
            LbVoucher.Text = TextBoxNum.Text;
            LabelUser.Text = txtNombre.Text;
            LabelCarnet.Text = Session["Carnet"].ToString();
            LabelCantidad.Text = (Int32.Parse(TextBoxTemp.Text)).ToString();
            index = 0;
            armartab();

/*Creacion del codigo de barra , como medio de seguridad en recibos */
            string barCode = TextBoxNum.Text;
            System.Web.UI.WebControls.Image imgBarCode = new
System.Web.UI.WebControls.Image();
            using (Bitmap bitMap = new Bitmap(barCode.Length * 40, 80))
            {
                using (Graphics graphics = Graphics.FromImage(bitMap))
                {
                    Font oFont = new Font("IDAutomationHC39M", 16);
                    PointF point = new PointF(2f, 2f);
                    SolidBrush blackBrush = new SolidBrush(Color.Black);
                    SolidBrush whiteBrush = new SolidBrush(Color.White);
                    graphics.FillRectangle(whiteBrush, 0, 0, bitMap.Width,
bitMap.Height);
                    graphics.DrawString("*" + "FEC" + "*", oFont, blackBrush,
point);
                }
                using (MemoryStream ms = new MemoryStream())
                {
                    bitMap.Save(ms, System.Drawing.Imaging.ImageFormat.Png);
                    byte[] byteImage = ms.ToArray();

                    Convert.ToBase64String(byteImage);
                    imgBarCode.ImageUrl = "data:image/png;base64," +
Convert.ToBase64String(byteImage);
                }
                plBarCode.Controls.Add(imgBarCode);
            }

            MPago.SetActiveView(VDetallePago);

```

```

    }

    else {

        MessageBoxNotification.SetMessage(App_Controls_MessageBox.MessageType.Warning, "Debe
        de llenar los campos solicitados");
        index = 0;
        armartab();

    }

}

```

## 28.4 CONFIRMACION DEL TRÁMITE

Después de haberse creado el previo del detalle de la transacción se hace el llamado a un (Procedimiento almacenado) que se encarga de almacenar la Información respectiva en las tablas requeridas.

Antes de dar por concluido el almacenamiento del trámite se realiza una confirmación previa antes de procesar dicho trámite.

```

protected void btnYes_Click(object sender, EventArgs e)
{
    /* Método que realiza la insercion de la nueva solicitud */
    RealizarTransaccion();
    if (userId > 0)
    {
        //Response.Write("<script LANGUAGE='JavaScript' >alert('Transaccion con
        Exito')</script>");

        MessageBoxNotification.SetMessage(App_Controls_MessageBox.MessageType.Success,
        "Transacción con Éxito");
        MPago.SetActiveView(VFill);

        BindDataUser();
        BindDataTramite();
        CargarCatalogo();
        TextBoxFechaSoli.Text = DateTime.Now.ToShortDateString();
        index = 0;
    }
}

```

```

        armartab();
        ddlTarjeta.SelectedIndex = 0;
        TextBoxPart1.Text = "";
        TextBoxPart2.Text = "";
        TextBoxPart3.Text = "";
        TextBoxPart4.Text = "";
        ddlyear.SelectedIndex = 0;
        ddlmes.SelectedIndex = 0;
        TextBoxCVV.Text = "";

        ddlCantidad.SelectedIndex = 0;
        ddlTipotramite.SelectedIndex = 0;
        ddlTipoDoc.SelectedIndex = 0;
        TextBoxPrecio.Text = "";
        TextBoxtiempo.Text = "";
    }
    else {

        MessageBoxNotification.SetMessage(App_Controls_MessageBox.MessageType.Error, "Error
        en la Transacción");
        MPago.SetActiveView(VFill);
        index = 0;
        armartab();
        ddlTarjeta.SelectedIndex = 0;
        TextBoxPart1.Text = "";
        TextBoxPart2.Text = "";
        TextBoxPart3.Text = "";
        TextBoxPart4.Text = "";
        ddlyear.SelectedIndex = 0;
        ddlmes.SelectedIndex = 0;
        TextBoxCVV.Text = "";

        ddlCantidad.SelectedIndex = 0;
        ddlTipotramite.SelectedIndex = 0;
        ddlTipoDoc.SelectedIndex = 0;
        TextBoxPrecio.Text = "";
        TextBoxtiempo.Text = "";
    }
    index = 0;
    armartab();
}
    }

```

## 28.5 INSERCIÓN DEL TRÁMITE

Para la Inserción de los trámites se utiliza el siguiente método, que hace el llamado a un procedimiento almacenado que recibe cierta cantidad de parámetros necesarios para llevar a cabo la culminación de la solicitud.

```

protected void RealizarTransaccion()
{

```

```

        string target = TextBoxPart1.Text + '-' + TextBoxPart2.Text + '-' +
        TextBoxPart3.Text + '-' + TextBoxPart4.Text;
        string carnet = Session["Carnet"].ToString();

        /*Nueva Conexion con la base de datos */
        Conexion conexion = new Conexion();
        using (SqlConnection con = Conexion.Conectar())
        {

        /*Nombre del método a ejecutar y sus parametros */

            using (SqlCommand cmd = new SqlCommand("Make_Process"))
            {
                cmd.CommandType = CommandType.StoredProcedure;
                cmd.Parameters.AddWithValue("@Nombre_Tra",
                (ddlTipoDoc.SelectedItem.ToString()));
                cmd.Parameters.AddWithValue("@NoCarnet", carnet);
                cmd.Parameters.AddWithValue("@Monto", float.Parse(LbMonto.Text));
                cmd.Parameters.AddWithValue("@cantidad",
                Int32.Parse(TextBoxTemp.Text));
                cmd.Parameters.AddWithValue("@IdDoc",
                Int32.Parse(ddlTipoDoc.SelectedValue));
                cmd.Parameters.AddWithValue("@IdBanco", 66);
                cmd.Parameters.AddWithValue("@IdTarjeta",
                Int32.Parse(ddlTarjeta.SelectedValue));
                cmd.Parameters.AddWithValue("@num_tarjeta", target);
                cmd.Connection = con;
                con.Open();
                userId = Convert.ToInt32(cmd.ExecuteScalar());
                con.Close();
            }

        }
    }
}

```

## 29. REPORTES

Para la generación de reportes se utilizó ReportViewer que se utiliza para hospedar informes en proyectos de ASP.NET.

### 29.1 REPORTE DE HISTORIAL DE TRÁMITES

Para este reporte se hace el llamado a un procedimiento almacenado que obtiene la información requerida según los parámetros que se hayan indicado. (Fig. 5)

The screenshot displays a web application interface. At the top, there is a form with a grid layout. The form contains the following elements:

- Fecha Inicio:** A text input field with a calendar icon to its right.
- Fecha Fin:** A text input field with a calendar icon to its right.
- Carnet:** A text input field.
- Estado:** A dropdown menu with the text "----Seleccione----" and a small square icon to its right.
- [Cb\_carne]:** A checkbox.
- [Cb\_estado]:** A checkbox.
- Aceptar:** A button with a circular icon and the text "Aceptar".

Below the form, there is a report viewer control labeled "rsweb:ReportViewer#ReportViewerHt". The viewer shows a single page of a report, indicated by "1 of 1" in the navigation bar. The report content area is currently blank and has a light blue background.

Fig. 5 Diseño del reporte de historial de trámites.

```
private void loadStoreProcedureTramite(int idreport)
{
    ReportViewerHt.Visible = true;

    /*Nueva Conexion con la base de datos*/
    Conexion conexion = new Conexion();
    using (SqlConnection con = Conexion.Conectar())
    {
        /*Nombre del procedimiento almacenado y sus parametros*/
        using (SqlCommand cmd = new SqlCommand("ReportTramite"))
        {
            cmd.CommandType = CommandType.StoredProcedure;
            cmd.Parameters.AddWithValue("@fechaI",
            Convert.ToDateTime(TextBoxHtI.Text));
            cmd.Parameters.AddWithValue("@fechaF",
            Convert.ToDateTime(TextBoxHtF.Text));
            cmd.Parameters.AddWithValue("@idTtramitante", TextBoxNumCar.Text);
        }
    }
}
```

```

        cmd.Parameters.AddWithValue("@estado",
Convert.ToInt32(ddlEstado.SelectedValue));
        cmd.Connection = con;
        con.Open();
        SqlDataReader sqlDR = cmd.ExecuteReader();
        DataTable dt = new DataTable();
        dt.Load(sqlDR);
        con.Close();
        DataSet ds = new DataSet();
        ds.Tables.Add(dt);
        ReportViewerHt.LocalReport.DataSources.Clear();

        var rpDtSource3 = new ReportDataSource("DataSet1", ds.Tables[0]);
        ReportViewerHt.LocalReport.DataSources.Add(rpDtSource3);

        ReportViewerHt.LocalReport.Refresh();
    }
}

```

1 of 1 Find | Next



**SISTEMA DE TRAMITES ACADEMICOS**

*Fecha Impresion: 22/08/2015 23:41:06*

Registro	Carnet	Fecha Solicitud	Fecha Entrega	Estado
<b>Carnet</b>				
1	P2012-0101	28/06/2015	10/07/2015	Concluido
<b>Carta de egresado</b>				
2	b2008-2828	05/07/2015	07/07/2015	Concluido
3	b2008-2828	07/07/2015	09/07/2015	En proceso
4	P2012-0101	07/07/2015	09/07/2015	En proceso
5	b2008-2828	24/07/2015	26/07/2015	Recibido
6	M2015-1515	13/08/2015	15/08/2015	Concluido
7	P2012-0101	13/08/2015	15/08/2015	Recibido

Fig. 6 Reporte de historial